IPSec のトラブルシューティング： debug コマンドの説明と使用

目次

概要

この資料は両方の IPsec 問題を解決するのに使用されるよくある **debug コマンドを** Cisco IOS 記述したものですか。 ソフトウェアおよび PIX/ASA。 このドキュメントでは、IPsec が設定済みであることを前提とします。 詳細については、『一般的な IPSec のエラー メッセージ』と『一般的な IPSec の問題』を参照してください。

IPSec VPN の問題に対する最も一般的なソリューションについては、『一般的な L2L およびリモート アクセス IPSec VPN のトラブルシューティング方法について』を参照してください。 接続および Cisco テクニカルサポートのトラブルシューティングを開始する前に試した方がよい一般的な手順のチェックリストが含まれています。

前提条件

要件

このドキュメントに関する固有の要件はありません。

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づくものです。

- **Cisco IOS ソフトウェア**IPSec フィーチャ セット56i - シングル Data Encryption Standard (DES; データ暗号規格) 機能を示します (Cisco IOS ソフトウェア リリース 11.2 以降) 。k2 - トリプル DES 機能を示します (Cisco IOS ソフトウェア リリース 12.0 以降) 。 トリプル DES は、Cisco 2600 シリーズ以降で使用できます。
- **PIX —** V5.0 およびそれ以降、アクティブになるために単一またはトリプルDES ライセンス キーが要求する。

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たものです。 このドキュメントで使用するすべてのデバイスは、クリアな (デフォルト) 設定で作業を開始しています。 ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

表記法

ドキュメント表記の詳細は、『シスコ テクニカル ティップスの表記法』を参照してください。

Cisco IOS ソフトウェアのデバッグ

このセクションの項目では、Cisco IOS ソフトウェアの debug コマンドについて解説します。 詳細については、『一般的な IPSec のエラー メッセージ』と『一般的な IPSec の問題』を参照してください。

show crypto isakmp sa

このコマンドは、ピア間で構築された Internet Security Association and Key Management Protocol（ISAKMP）セキュリティ アソシエーション（SA）を表示します。

```
dst      src        state     conn-id     slot
12.1.1.2  12.1.1.1   QM_IDLE    1           0
```

show crypto ipsec sa

このコマンドは、ピア間で構築された IPSec SA を表示します。 ネットワーク 20.1.1.0 と 10.1.1.0 の間を流れるトラフィックに対して、12.1.1.1 と 12.1.1.2 との間に暗号化されたトンネルが構築されます。 着信側および発信側で構築された 2 つの Encapsulating Security Payload（ESP）SA を確認できます。 Authentication Header（AH; 認証ヘッダー）SA がないため、AH は使用されていません。

show crypto ipsec sa コマンドの出力例を次に示します。

```
  interface: FastEthernet0
    Crypto map tag: test, local addr. 12.1.1.1
   local  ident (addr/mask/prot/port): (20.1.1.0/255.255.255.0/0/0)
   remote ident (addr/mask/prot/port): (10.1.1.0/255.255.255.0/0/0)
   current_peer: 12.1.1.2
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 7767918, #pkts encrypt: 7767918, #pkts digest 7767918
    #pkts decaps: 7760382, #pkts decrypt: 7760382, #pkts verify 7760382
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0,
    #pkts decompress failed: 0, #send errors 1, #recv errors 0
     local crypto endpt.: 12.1.1.1, remote crypto endpt.: 12.1.1.2
     path mtu 1500, media mtu 1500
     current outbound spi: 3D3
     inbound esp sas:
      spi: 0x136A010F(325714191)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 3442, flow_id: 1443, crypto map: test
        sa timing: remaining key lifetime (k/sec): (4608000/52)
        IV size: 8 bytes
        replay detection support: Y
     inbound ah sas:
     inbound pcp sas:
inbound pcp sas:
outbound esp sas:
   spi: 0x3D3(979)
    transform: esp-3des esp-md5-hmac ,
    in use settings ={Tunnel, }
    slot: 0, conn id: 3443, flow_id: 1444, crypto map: test
    sa timing: remaining key lifetime (k/sec): (4608000/52)
    IV size: 8 bytes
    replay detection support: Y
outbound ah sas:
```

```
outbound pcp sas:
```

show crypto engine connection active

このコマンドは構築される毎フェーズ 2 SA および送信 される トラフィック量を示したものです。 フェーズ 2 (セキュリティ アソシエーション) SA は単方向であるため、各 SA には一方向のトラフィックだけが表示されます (暗号化は発信側、復号化は着信側です) 。

debug crypto isakmp

debug crypto isakmp コマンドの出力例を次に示します。

```
processing SA payload. message ID = 0
Checking ISAKMP transform against priority 1 policy
encryption DES-CBC
      hash SHA
default group 2
auth pre-share
life type in seconds
life duration (basic) of 240
atts are acceptable. Next payload is 0
processing KE payload. message ID = 0
processing NONCE payload. message ID = 0
processing ID payload. message ID = 0
SKEYID state generated
processing HASH payload. message ID = 0
SA has been authenticated
processing SA payload. message ID = 800032287
```

debug crypto ipsec

このコマンドは、発信元および送信先の IPSec トンネル エンドポイントを表示します。 src_proxy および dest_proxy はクライアント サブネットです。 2 つの "sa created" メッセージが、方向ごとに 1 つずつ表示されます （ ESP および AH を実行すると、メッセージが 4 つ表示されます) 。

debug crypto ipsec コマンドの出力例を次に示します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
```

```
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

エラー メッセージの例

ここでは、次の各 debug コマンドで生成されたエラー メッセージの例を示します。

- debug crypto ipsec
- debug crypto isakmp
- debug crypt engine

Replay Check Failed (リプレイ チェックに失敗)

「Replay Check Failed」エラーの出力例を次に示します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
```

```
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

このエラーは、伝送メディアでリオーダーした結果（特にパラレル パスが存在する場合）によるものです。または、負荷時に大きいパケットと小さいパケットに対して Cisco IOS の内部で行われるパケット処理の不適切なパスによるものです。 これを反映するために、トランスフォーム セットを変更します。 reply check は、transform-set esp-md5-hmac が有効にされているときだけ見られます。 このエラー メッセージを抑制するには、esp-md5-hmac をディセーブルにして、暗号化のみを行います。 Cisco Bug ID CSCdp19680（登録ユーザ専用）を参照してください。

IPsec リプレイ防止ウィンドウを設定する方法についての情報に関しては IPsec リプレイ防止ウィンドウを設定する方法を参照して下さい: **拡張とディセーブル化**

QM FSM エラー

IPsec L2L VPN トンネルが PIX ファイアウォールまたは ASA で確立されず、*QM FSM* エラー メッセージが表示されます。

1 つの原因としては、プロキシ アイデンティティ（対象トラフィック、Access Control List（ACL; アクセス コントロール リスト）、クリプト ACL など）が両端で一致していない場合が考えられます。 両方のデバイス上でコンフィギュレーションを確認し、暗号化 ACL が一致していることを確認します。

もう一つの考えられる原因はトランスフォームによって設定 される パラメータの組合わせを誤まることです。 両端で、VPNゲートウェイが正確のと設定 される同じトランスフォームを同じパラメータ使用することを確かめて下さい。

Invalid Local Address（無効なローカル アドレス）

エラー メッセージの出力例を次に示します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
```

```
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

このエラー メッセージは、次の 2 つの一般的な問題のいずれかが原因です。

- crypto map map-name local-address interface-id コマンドにより、ルータが特定のアドレスの使用を強制されたため、ルータが識別情報として正しくないアドレスを使用した。
- crypto map が間違ったインターフェイスに適用されているか、またはまったく適用されていない。 設定をチェックし、crypto map が正しいインターフェイスに適用されていることを確認します。

X.X.X.X 正常性チェックからの IKE メッセージは失敗するか、または不正です

この debug エラーは、ピア同士の事前共有キーが一致しない場合に表示されます。 この問題を修正するには、両方の側の事前共有キーを確認します。

```
Checking IPSec proposal 1transform 1, ESP_DES
```

```
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

メインモードの処理はピアと失敗しました

Main Mode (メイン モード) エラー メッセージの例を次に示します。 メイン モードの失敗は、フェーズ 1 ポリシーが両方の側で一致しないことを示しています。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
```

```
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

show crypto isakmp sa コマンドは、ISAKMP SA が MM_NO_STATE にあることを示します。 これは、メイン モードが失敗したことも意味します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
```

```
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

フェーズ 1 ポリシーが両方のピアにあることを確認し、すべての属性が一致していることを確かめます。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
```

```
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

Proxy identities not supported (プロキシ ID がサポートされていない)

IPSec トラフィックのアクセス リストが一致していない場合、デバッグにはこのメッセージが表示されます。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
```

```
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

各ピアのアクセス リストは互いにミラーになっている（すべてのエントリが反転している）必要があります。 次の例は、この点について説明しています。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

Transform Proposal Not Supported（トランスフォーム プロポーザルがサポートされていない）

このメッセージは、フェーズ 2（IPSec）が両方の側で一致しない場合に表示されます。 このメッセージが最もよく発生するのは、トランスフォーム セット内に不一致や非互換性が存在する場

合です。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

トランスフォーム セットが両方の側で一致することを確認します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
```

```
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

リモートピアでの No cert and no keys

このメッセージは、ルータに設定されたピア アドレスが間違っているか、または変更されたことを示しています。 ピア アドレスが正しいこと、およびアドレスが到達可能であることを確認します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
```

```
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

Peer Address X.X.X.X Not Found (ピア アドレス X.X.X.X が見つからない)

通常、このエラー メッセージは、対応する VPN 3000 コンセントレータのエラー メッセージ「Message: No proposal chosen(14)」とともに表示されます。 このエラーは、接続がホストツーホストになった結果を示しています。 ルータ コンフィギュレーションでは、IPSec プロポーザルの順序が、ルータ用に選択されたプロポーザルがピアではなくアクセス リストに一致するようになっています。 アクセス リストにはそれよりも大きなネットワークが指定されていて、トラフィックと交差しているホストはこれに含まれています。 これを修正するには、コンセントレータからルータまでのこの接続に対応するルータ プロポーザルが先に行に現れるようにします。 これにより、ルータ プロポーザルが特定のホストに先に一致します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
```

```
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

IPsec Packet has Invalid SPI（IPSec パケットに無効な SPI がある）

エラー メッセージの出力例を次に示します。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
```

```
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

受信した IPSec パケットには、Security Associations Database（SADB; セキュリティ アソシエーション データベース）に存在しない Security Parameters Index（SPI; セキュリティ パラメータ インデックス）が指定されています。 これは次の原因による一時的な状態である可能性があります。

- IPSec ピア間の Security Association（SA）のエージングにわずかな相違がある
- ローカル SA がクリアされている
- IPSec ピアによって誤ったパケットが送信された

これは、攻撃の可能性もあります。

推奨処置：ピアでは、ローカル SA がクリアされたことが認識されない場合があります。 ローカル ルータから新しい接続が確立されると、その後は、2 つのピアでの正常な再構築が可能です。 その他の場合、問題が短期間で収束しない場合は、新しい接続を確立するか、またはピアの管理者に連絡してください。

IPSEC(initialize_sas): 無効なプロキシID

エラー「21:57:57: `IPSEC(initialize_sas)`: invalid proxy IDs」は、受信したプロキシ ID が、アクセス リストを基準として設定されているプロキシ ID と一致しないことを示しています。 両方が一致していることを確かめるには、**debug** コマンドによる出力を確認します。

プロポーザル要求の **debug** コマンドの出力では、対応する access-list 103 permit ip 10.1.1.0 0.0.0.255 20.1.1.0 0.0.0.255 が一致していません。 アクセス リストは、一方の端ではネットワーク固有であり、他方の端ではホスト固有です。

```
Checking IPSec proposal 1transform 1, ESP_DES
attributes in transform:
encaps is 1
SA life type in seconds
SA life duration (basic) of 3600
SA life type in kilobytes
SA life duration (VPI) of 0x0 0x46 0x50 0x0
HMAC algorithm is SHA
atts are acceptable.
Invalid attribute combinations between peers will show up as "atts
  not acceptable".
IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 20.1.1.0/0.0.0.16/0/0,
```

```
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 0s and 0kb,
      spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 203563166 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 12.1.1.2 to 12.1.1.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.2, SRC= 12.1.1.1,
      dest_proxy= 10.1.1.0/255.255.255.0/0/0,
      src_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xC22209E(203563166), conn_id= 3,
              keysize=0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 12.1.1.2, dest= 12.1.1.1,
      src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 20.1.1.0/255.255.255.0/0/0,
      protocol= ESP, transform= esp-des esp-sha-hmac
      lifedur= 3600s and 4608000kb,
      spi= 0xDED0AB4(233638580), conn_id= 6,
              keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xB9D0109(194838793),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
 IPSEC(create_sa): sa created,
      (sa) sa_dest= 12.1.1.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

Reserved Not Zero on Payload 5 (ペイロード 5 で Reserved が 0 ではない)

これは、ISAKMP キーが一致しないことを示します。 正確さを確実にするには、キーの再生成かリセットを行います。

Hash Algorithm Offered does not Match Policy (提供されるハッシュ アルゴリズムがポリシーと一致しない)

設定された ISAKMP ポリシーがリモート ピアによって提示されたポリシーと一致しない場合、ルータは 65535 のデフォルト ポリシーを試行します。 それも一致しない場合は、ISAKMP ネゴシエーションが失敗します。 ユーザは、ルータ上でエラー メッセージ「Hash algorithm offered does not match policy! 」または「Encryption algorithm offered does not match policy! 」を受信します。

```
=RouterA=
3d01h: ISAKMP (0:1): processing SA payload. message ID = 0
3d01h: ISAKMP (0:1): found peer pre-shared key matching 209.165.200.227
ISAKMP (0:1): Checking ISAKMP transform 1 against priority 1 policy
ISAKMP:      encryption 3DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 1
ISAKMP:      auth pre-share
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of  0x0 0x1 0x51 0x80
ISAKMP (0:1): Hash algorithm offered does not match policy!
ISAKMP (0:1): atts are not acceptable. Next payload is 0
```

```
=RouterB=
ISAKMP (0:1): Checking ISAKMP transform 1 against priority 65535 policy
ISAKMP:      encryption 3DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 1
ISAKMP:      auth pre-share
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of  0x0 0x1 0x51 0x80
ISAKMP (0:1): Encryption algorithm offered does not match policy!
ISAKMP (0:1): atts are not acceptable. Next payload is 0
ISAKMP (0:1): no offers accepted!
ISAKMP (0:1): phase 1 SA not acceptable!
```

HMAC Verification Failed (HMAC の確認に失敗)

IPSec パケットで Hash Message Authentication Code (HMAC) の確認が失敗すると、このエラー メッセージが報告されます。 これは通常、パケットが破損している場合に起こります。

```
Sep 22 11:02:39 131.203.252.166 2435:
Sep 22 11:02:39: %MOTCR-1-ERROR:motcr_crypto_callback() motcr return failure
Sep 22 11:02:39 131.203.252.166 2436:
Sep 22 11:02:39: %MOTCR-1-PKTENGRET_ERROR: MOTCR PktEng Return Value = 0x20000,
                 PktEngReturn_MACMiscompare
```

このエラー メッセージがときどき表示されるのであれば、無視できます。 しかし、たびたび繰り返される場合は、実際に何がパケットを破損させているか調査する必要があります。 この原因は、暗号アクセラレータの欠陥による場合もあります。

Remote Peer Not Responding (リモート ピアが応答しない)

このエラー メッセージは、トランスフォーム セットの不一致がある場合に表示されます。 両方のピアで一致したトランスフォーム セットが設定されていることを確認してください。

すべての IPSec SA 提案は受け入れられないと思いました

このエラーメッセージはフェーズ 2 IPSecパラメータがローカル および リモートサイトの間で組合わせを誤まられるとき表示されます。 一致する、正常な VPN が確立するようにこの問題を解決するために、設定 される トランスフォームの同じパラメータを規定 して下さい。

Packet Encryption/Decryption Error (パケットの暗号化/復号化エラー)

エラー メッセージの出力例を次に示します。

```
Sep 22 11:02:39 131.203.252.166 2435:
Sep 22 11:02:39: %MOTCR-1-ERROR:motcr_crypto_callback() motcr return failure
Sep 22 11:02:39 131.203.252.166 2436:
Sep 22 11:02:39: %MOTCR-1-PKTENGRET_ERROR: MOTCR PktEng Return Value = 0x20000,
                 PktEngReturn_MACMiscompare
```

このエラー メッセージの原因としては、次のいずれかが考えられます。

- フラグメンテーション : フラグメント化された crypto パケットのプロセス スイッチングにより、プロセス スイッチングされたパケットよりも前に、ファスト スイッチングされたパケットが VPN カードに強制的に送信される。 プロセス スイッチングされたパケットの前に、ファスト スイッチングされた十分な数のパケットが処理されると、プロセス スイッチングされたパケットの ESP または AH のシーケンス番号が期限切れになるため、それらのパケットが VPN カードに着信したとき、該当するシーケンス番号がリプレイ ウィンドウの範囲外になり

ます。 これにより、使用するカプセル化によって異なりますが、AH または ESP シーケンス番号エラー (それぞれ 4615 および 4612) が発生します。
- *キャッシュ エントリの期限切れ* : ファスト スイッチングされたキャッシュ エントリが期限切れになり、キャッシュ ミスが発生した先頭パケットがプロセス スイッチングされた場合にも、このエラー メッセージが発生することがあります。

回避策

1. 3DES トランスフォーム セット上のすべてのタイプの認証をオフにし、ESP-DES/3DES を使用します。 これにより、認証/再生防止保護が無効になり、順序の不正な (混合の) IPsec トラフィックに関連したパケット破棄エラー *%HW_VPN-1-HPRXERR: Hardware VPN0/2: Packet Encryption/Decryption error, status=4615* が回避されます。
2. 前述の項番 1 の理由に対して実際に適用できる回避策の 1 つ は、着信ストリームの Maximum Transmission Unit (MTU; 最大伝送ユニット) のサイズを 1400 バイト未満にする方法です。 MTU のサイズを 1400 バイト未満にするには、次のコマンドを入力します。`Sep 22 11:02:39 131.203.252.166 2435:`
   `Sep 22 11:02:39: %MOTCR-1-ERROR:motcr_crypto_callback() motcr return failure`
   `Sep 22 11:02:39 131.203.252.166 2436:`
   `Sep 22 11:02:39: %MOTCR-1-PKTENGRET_ERROR: MOTCR PktEng Return Value = 0x20000, PktEngReturn_MACMiscompare`
3. AIM カードを無効にします。
4. ルータ インターフェイス上で/CEF のスイッチングをオフにします。 ファスト スイッチングを削除するには、インターフェイス設定モードで次のコマンドを実行します。`Sep 22 11:02:39 131.203.252.166 2435:`
   `Sep 22 11:02:39: %MOTCR-1-ERROR:motcr_crypto_callback() motcr return failure`
   `Sep 22 11:02:39 131.203.252.166 2436:`
   `Sep 22 11:02:39: %MOTCR-1-PKTENGRET_ERROR: MOTCR PktEng Return Value = 0x20000, PktEngReturn_MACMiscompare`

ESP シーケンス失敗によるパケット受信エラー

エラーメッセージの例はここにあります:

```
Sep 22 11:02:39 131.203.252.166 2435:
Sep 22 11:02:39: %MOTCR-1-ERROR:motcr_crypto_callback() motcr return failure
Sep 22 11:02:39 131.203.252.166 2436:
Sep 22 11:02:39: %MOTCR-1-PKTENGRET_ERROR: MOTCR PktEng Return Value = 0x20000,
                 PktEngReturn_MACMiscompare
```

このエラーメッセージは通常これらの可能性のある状態の 1 つを示します:

- IPsec 暗号化されたパケットは不適切に設定された QoS メカニズムが理由で暗号化ルータによって順番が異なる転送されます。
- 復号化ルータによって受信される IPsecパケットは中間デバイスで追加注文するパケットが順番が異なる な原因です。
- 受信された IPsecパケットはフラグメント化し、認証確認および復号化の前に再組立てを必要とします。

回避策

1. 暗号化するか、または中間ルータの IPSecトラフィックのための QoS をディセーブルにして下さい。
2. 暗号化ルータの IPsec 前フラグメンテーションを有効に して下さい。`Router(config-if)#crypto ipsec fragmentation before-encryption`
3. フラグメント化する必要がないサイズに MTU 値を設定 して下さい。

```
Router(config)#interface type [slot_#/]port_#
Router(config-if)#ip mtu MTU_size_in_bytes
```

4. そのトレインの最新の利用可能 な 安定 した イメージに IOSイメージをアップグレードして下さい。

注: 中断 されるべきそのインターフェイスで終わったあらゆるルータ インターフェイスの MTU サイズをすべてのトンネルを引き起こします変更するにより。 スケジュールされたダウンタイムの間にこの回避策を完了することを計画して下さい。

7600 シリーズ ルータの VPN トンネルを確立することを試みるエラー

このエラーは 7600 シリーズ ルータの VPN トンネルを確立することを試みるとき受け取られます:

```
Router(config-if)#ip mtu MTU_size_in_bytes
```

このエラーはソフトウェア暗号化が 7600 シリーズ ルータでサポートされないので発生します。 7600 シリーズ ルータは IPsec SPA ハードウェアなしでは IPSecトンネル 終了をサポートしません。 VPN は 7600 年のルータの IPSEC-SPA カードでだけサポートされます。

PIX のデバッグ

show crypto isakmp sa

このコマンドは、ピア間で構築された ISAKMP SA を表示します。

```
Router(config-if)#ip mtu MTU_size_in_bytes
```

show crypto isakmp sa コマンドの出力では、状態は常に QM_IDLE である必要があります。 状態が MM_KEY_EXCH である場合は、設定された事前共有キーが正しくないか、ピアの IP アドレスが異なっていることを意味します。

```
PIX(config)#show crypto isakmp sa
Total     : 2
Embryonic : 1
        dst               src        state     pending     created
 192.168.254.250   10.177.243.187    MM_KEY_EXCH   0           0
```

この問題は、正しい IP アドレスか事前共有キーを設定することで修正できます。

show crypto ipsec sa

このコマンドは、ピア間で構築された IPSec SA を表示します。 ネットワーク 20.1.1.0 と 10.1.1.0 の間を流れるトラフィックに対して、12.1.1.1 と 12.1.1.2 との間に暗号化されたトンネルが構築されます。 着信側および発信側で構築された 2 つの ESP SA を確認できます。 AH SA がないため、AH は使用されていません。

show crypto ipsec sa コマンドの例は、次の出力に示されています。

```
interface: outside
    Crypto map tag: vpn, local addr. 12.1.1.1
   local  ident (addr/mask/prot/port): (20.1.1.0/255.255.255.0/0/0)
   remote ident (addr/mask/prot/port): (12.1.1.2/255.255.255.255/0/0)
   current_peer: 10.2.1.1
   dynamic allocated peer ip: 12.1.1.2
     PERMIT, flags={}
    #pkts encaps: 345, #pkts encrypt: 345, #pkts digest 0
```

```
    #pkts decaps: 366, #pkts decrypt: 366, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0,
    #pkts decompress failed: 0, #send errors 0, #recv errors 0
     local crypto endpt.: 12.1.1.1, remote crypto endpt.: 12.1.1.2
     path mtu 1500, ipsec overhead 56, media mtu 1500
     current outbound spi: 9a46ecae
     inbound esp sas:
      spi: 0x50b98b5(84646069)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 1, crypto map: vpn
        sa timing: remaining key lifetime (k/sec): (460800/21)
        IV size: 8 bytes
        replay detection support: Y
     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x9a46ecae(2588339374)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2, crypto map: vpn
        sa timing: remaining key lifetime (k/sec): (460800/21)
        IV size: 8 bytes
        replay detection support: Y
    outbound ah sas:
```

debug crypto isakmp

このコマンドでは IPSec 接続に関するデバッグ情報が表示され、両端で互換性がないために拒否された最初の属性セットが示されます。 2 回目の照合試行（DES および Secure Hash Algorithm（SHA; 安全なハッシュアルゴリズム）の代わりにトリプル DES を試行）は受け入れられ、ISAKMP SA が構築されます。 このデバッグはローカル プールからの IP アドレス（10.32.8.1）を受け入れるダイヤルアップ クライアントからも出力されます。 ISAKMP SA が構築されると、IPSec 属性がネゴシエートされ、受け入れ可能と判断されます。 これにより、PIX は次のように IPSec SA を設定します。

debug crypto isakmp コマンドの出力例を次に示します。

```
crypto_isakmp_process_block: src 12.1.1.1, dest 12.1.1.2
OAK_AG exchange
ISAKMP (0): processing SA payload. message ID = 0
ISAKMP (0): Checking ISAKMP transform 1 against priority 1 policy
ISAKMP:      encryption DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 1
ISAKMP:      auth pre-share
ISAKMP (0): atts are not acceptable. Next payload is 3
ISAKMP (0): Checking ISAKMP transform 3 against priority 1 policy
ISAKMP:      encryption 3DES-CBC
ISAKMP:      hash SHA
ISAKMP:      default group 1
ISAKMP:      auth pre-share
ISAKMP (0): atts are acceptable. Next payload is 3
ISAKMP (0): processing KE payload. message ID = 0
ISAKMP: Created a peer node for 12.1.1.2
OAK_QM exchange
ISAKMP (0:0): Need config/address
ISAKMP (0:0): initiating peer config to 12.1.1.2. ID = 2607270170 (0x9b67c91a)
return status is IKMP_NO_ERROR
```

```
crypto_isakmp_process_block: src 12.1.1.2, dest 12.1.1.1
ISAKMP_TRANSACTION exchange
ISAKMP (0:0): processing transaction payload from 12.1.1.2.
   message ID = 2156506360
ISAKMP: Config payload CFG_ACK
ISAKMP (0:0): peer accepted the address!
ISAKMP (0:0): processing saved QM.
oakley_process_quick_mode:
OAK_QM_IDLE
ISAKMP (0): processing SA payload. message ID = 818324052
ISAKMP : Checking IPSec proposal 1
ISAKMP: transform 1, ESP_DES
ISAKMP:   attributes in transform:
ISAKMP:      authenticator is HMAC-MD5
ISAKMP:      encaps is 1
IPSEC(validate_proposal): transform proposal
   (prot 3, trans 2, hmac_alg 1) not supported
ISAKMP (0): atts not acceptable. Next payload is 0
ISAKMP : Checking IPSec proposal 2
ISAKMP: transform 1, ESP_3DES
ISAKMP:   attributes in transform:
ISAKMP:      authenticator is HMAC-MD5
ISAKMP:      encaps is 1
ISAKMP (0): atts are acceptable.
ISAKMP (0): processing NONCE payload. message ID = 818324052
ISAKMP (0): processing ID payload. message ID = 81
ISAKMP (0): ID_IPV4_ADDR src 10.32.8.1 prot 0 port 0
ISAKMP (0): processing ID payload. message ID = 81
ISAKMP (0): ID_IPV4_ADDR dst 12.1.1.1 prot 0 port 0
INITIAL_CONTACTIPSEC(key_engine): got a queue event...
```

debug crypto ipsec

このコマンドでは、IPSec 接続に関する **デバッグg** 情報が表示されます。

```
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 0xd532efbd(3576885181) for SA
        from  12.1.1.2  to  12.1.1.1  for prot 3
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 12.1.1.2, dest 12.1.1.1
OAK_QM exchange
oakley_process_quick_mode:
OAK_QM_AUTH_AWAIT
ISAKMP (0): Creating IPSec SAs
        inbound SA from  12.1.1.2  to  12.1.1.1
           (proxy 10.32.8.1 to  12.1.1.1.)
        has spi 3576885181 and conn_id 2 and flags 4
        outbound SA from  12.1.1.1  to  12.1.1.2
           (proxy 12.1.1.1 to 10.32.8.1)
        has spi 2749108168 and conn_id 1 and flags 4IPSEC(key_engine):
           got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 12.1.1.1, src= 12.1.1.2,
    dest_proxy= 12.1.1.1/0.0.0.0/0/0 (type=1),
    src_proxy= 10.32.8.1/0.0.0.0/0/0 (type=1),
    protocol= ESP, transform= esp-3des esp-md5-hmac ,
    lifedur= 0s and 0kb,
    spi= 0xd532efbd(3576885181), conn_id= 2, keysize= 0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) src= 12.1.1.1, dest= 12.1.1.2,
    src_proxy= 12.1.1.1/0.0.0.0/0/0 (type=1),
    dest_proxy= 10.32.8.1/0.0.0.0/0/0 (type=1),
    protocol= ESP, transform= esp-3des esp-md5-hmac ,
```

```
    lifedur= 0s and 0kb,
    spi= 0xa3dc0fc8(2749108168), conn_id= 1, keysize= 0, flags= 0x4
return status is IKMP_NO_ERROR
```

ルータと VPN Client 間の一般的な問題

VPN トンネル外部のサブネットにアクセスできない： スプリット トンネリング

次のルータ コンフィギュレーションの出力例は、VPN 接続のためのスプリット トンネリングを有効にする方法を示しています。 access list 150 コマンドは、crypto isakmp client configuration group hw-client-groupname コマンドの中で設定されているグループに関連付けられます。 これにより、Cisco VPN Client はルータを使用して、VPN トンネルの構成要素ではない追加のサブネットにアクセスできます。 これは、IPSec 接続のセキュリティを損なうことなく行えます。 トンネルは 172.168.0.128 ネットワーク上に形成されます。 **access list 150** コマンドで定義されていないデバイス (インターネットなど) へのトラフィック フローは暗号化されません。

```
!
crypto isakmp client configuration group hw-client-groupname
 key hw-client-password
 dns 172.168.0.250 172.168.0.251
 wins 172.168.0.252 172.168.0.253
 domain cisco.com
 pool dynpool
 acl 150
!
!
access-list 150 permit ip 172.168.0.128 0.0.0.127 any
!
```

PIX と VPN Client 間の一般的な問題

次のセクションは、VPN Client 3.x.を使用して IPSec に PIX を設定する際に遭遇する一般的な問題について説明しています。 PIX の設定例は、バージョン 6.x に基づくものです。

トンネル確立後にトラフィックが流れない： PIX 背後の内部ネットワークに PING が通らない

この問題はルーティングに関連する一般的な問題です。 内部にあり、なおかつ同じサブネットに直接接続されていないネットワークに対応する経路が PIX にあることを確認します。 また、内部ネットワークには、クライアント アドレス プール内のアドレスごとに、PIX に戻る経路がある必要があります。

出力例を次に示します。

```
!
crypto isakmp client configuration group hw-client-groupname
 key hw-client-password
 dns 172.168.0.250 172.168.0.251
 wins 172.168.0.252 172.168.0.253
 domain cisco.com
 pool dynpool
 acl 150
!
!
access-list 150 permit ip 172.168.0.128 0.0.0.127 any
!
```

トンネルのアップ後、ユーザがインターネットをブラウズできない： スプリット トンネリング

この問題の最も一般的な原因は、VPN Client から PIX への IPSec トンネルによって、すべてのトラフィックがトンネルを通じて PIX ファイアウォールに送られることです。 PIX の機能では、トラフィックを受信したインターフェイスにそのトラフィックを送信し戻すことは許可されていません。 そのため、インターネット宛てのトラフィックは動作しません。 この問題を修正するには、**split tunneling** コマンドを使用します。 この修正は、ただ 1 つの特定のトラフィックだけをトンネル経由で送信し、残りのトラフィックはトンネル経由ではなく直接インターネットに送る、という考え方に基づいています。

```
vpngroup vpn3000 split-tunnel 90
access-list 90 permit ip 10.1.1.0 255.255.255.0 10.1.2.0 255.255.255.0
access-list 90 permit ip 172.16.0.0 255.255.0.0 10.1.2.0 255.255.255.0
```

注: vpngroup vpn3000 split-tunnel 90 コマンドにより、**アクセス リスト番号 90** による分割トンネリングが可能になります。 access-list 90 コマンドでは、どのトラフィックをトンネル経由で流すかが定義され、残りのトラフィックはアクセス リストの最後で拒否されます。 PIX で Network Address Translation（NAT; ネットワーク アドレス変換）を拒否する場合にも、同じアクセス リストを設定する必要があります。

トンネルのアップ後、特定のアプリケーションが動作しない： クライアントでの MTU 調整

トンネルが確立された後、ユーザが PIX ファイアウォールの背後にあるネットワーク上のマシンに PING できても、Microsoft Outlook などの特定のアプリケーションを使用できないことがあります。 よく見られる問題は、パケットの Maximum Transfer Unit（MTU; 最大伝送ユニット）のサイズです。 オリジナルパケットに付加される IPSecヘッダーは 50バイトから 60バイトまである場合もあります。 パケットのサイズが 1500（インターネットでのデフォルト）を超えた場合、各デバイスでパケットをフラグメント化する必要があります。 IPsec ヘッダーが追加されても、サイズはまだ IPSec の最大値である 1496 未満です。

show interface コマンドでは、構内のルータまたはアクセス可能なルータ上の特定のインターフェイスの MTU が示されます。 発信元から宛先までのすべてのパスの MTU を判別するために、送信されたデータグラムが MTU より多い場合にこのエラー メッセージが発信元に送り返されるように、さまざまなサイズのデータグラムが DF（Don't Fragment）ビットを設定して送信されます。

```
vpngroup vpn3000 split-tunnel 90
access-list 90 permit ip 10.1.1.0 255.255.255.0 10.1.2.0 255.255.255.0
access-list 90 permit ip 172.16.0.0 255.255.0.0 10.1.2.0 255.255.255.0
```

この出力例では、IP アドレスが 10.1.1.2 と 172.16.1.56 のホスト間でパスの MTU を見つける例が示されています。

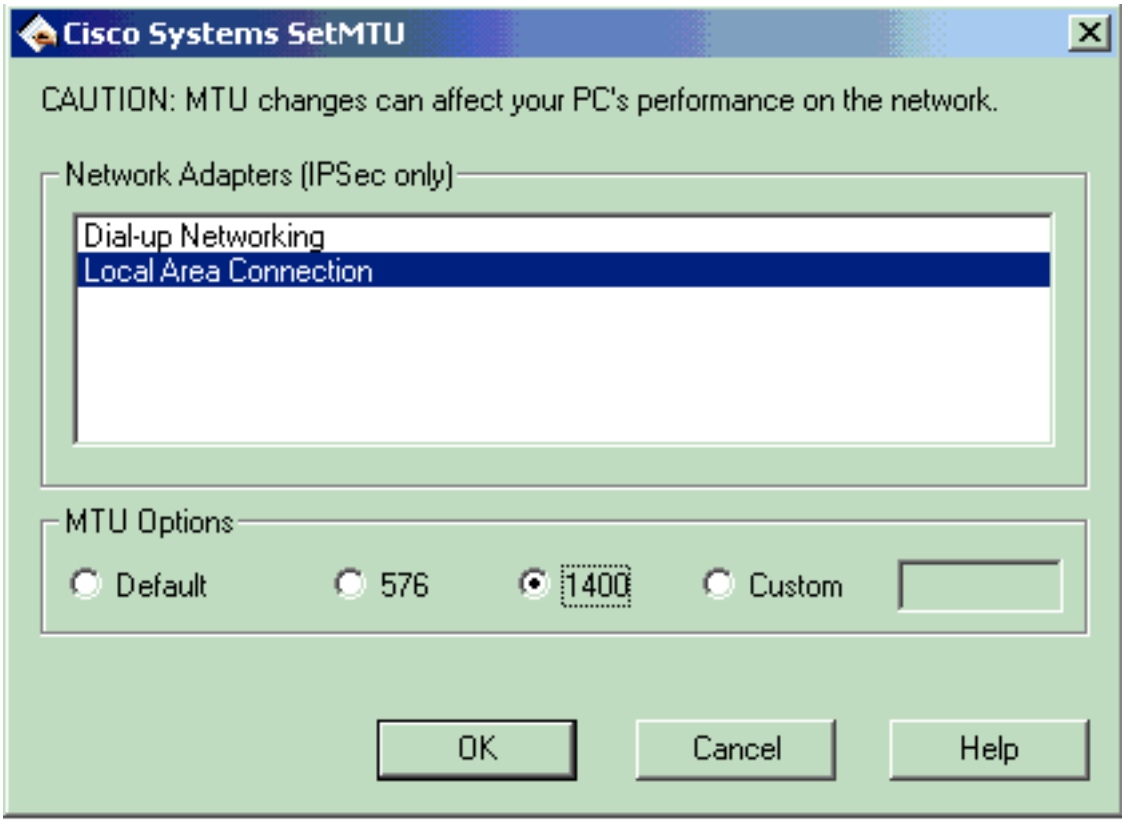
```
Router#debug ip icmp
ICMP packet debugging is on

!--- Perform an extended ping. Router#ping
Protocol [ip]:
Target IP address: 172.16.1.56
Repeat count [5]:
Datagram size [100]: 1550
Timeout in seconds [2]:
```

```
!--- Make sure you enter y for extended commands. Extended commands [n]: y
Source address or interface: 10.1.1.2
Type of service [0]:

!--- Set the DF bit as shown. Set DF bit in IP header? [no]: y
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 1550-byte ICMP Echos to 172.16.1.56, timeout is 2 seconds:

2w5d: ICMP: dst (172.16.1.56): frag. needed and DF set.
2w5d: ICMP: dst (172.16.1.56): frag. needed and DF set.
2w5d: ICMP: dst (172.16.1.56): frag. needed and DF set.
2w5d: ICMP: dst (172.16.1.56): frag. needed and DF set.
2w5d: ICMP: dst (172.16.1.56): frag. needed and DF set.
Success rate is 0 percent (0/5)

!--- Reduce the datagram size further and perform extended ping again. Router#ping
Protocol [ip]:
Target IP address: 172.16.1.56
Repeat count [5]:
Datagram size [100]: 1500
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.1.1.2
Type of service [0]:
Set DF bit in IP header? [no]: y
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 172.16.1.56, timeout is 2 seconds:
!!!!!
2w5d: ICMP: echo reply rcvd, src 172.16.1.56, dst 10.1.1.2
2w5d: ICMP: echo reply rcvd, src 172.16.1.56, dst 10.1.1.2
2w5d: ICMP: echo reply rcvd, src 172.16.1.56, dst 10.1.1.2
2w5d: ICMP: echo reply rcvd, src 172.16.1.56, dst 10.1.1.2
2w5d: ICMP: echo reply rcvd, src 172.16.1.56, dst 10.1.1.2

Success rate is 100 percent (5/5), round-trip min/avg/max = 380/383/384 ms
```

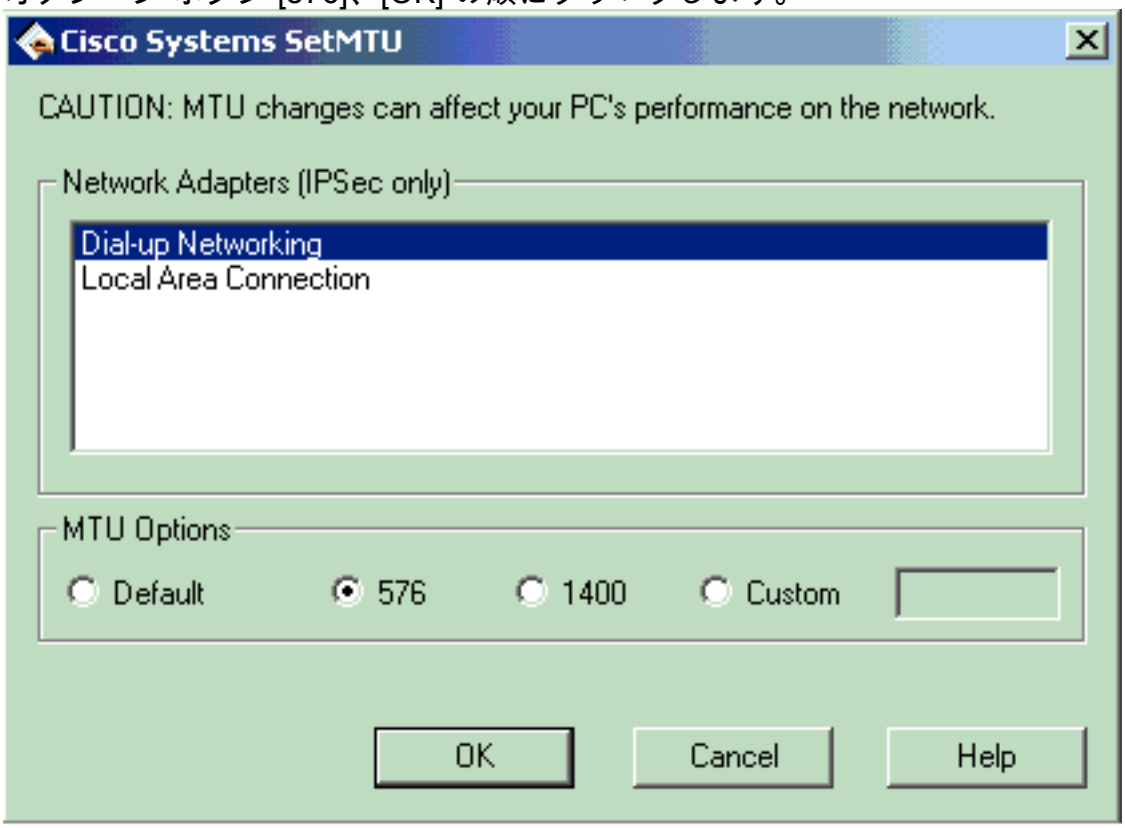
注: VPN Client には MTU 調整ユーティリティが付属しており、ユーザはこれを使用して Cisco VPN Client の MTU を調整できます。 PPP over Ethernet（PPPoE）クライアント ユーザの場合は、PPPoE アダプタの MTU を調整します。

注: VPN Client の MTU ユーティリティを調整するには、次の手順を実行します。

1. [Start] > [Programs] > [Cisco System VPN Client] > [Set MTU] の順に選択します。
2. [Local Area Connection] を選択し、[1400] オプション ボタンをクリックします。
3. [OK] をクリックします。



4. ステップ 1 を繰り返し、[Dial-up Networking] を選択します。
5. オプション ボタン [576]、[OK] の順にクリックします。



sysopt コマンドが設定されていない

PIX の IPSec コンフィギュレーションで **sysopt connection permit-ipsec** コマンドを使用し、**conduit** や **access-list** コマンド文のチェックなしで IPSec トラフィックが PIX ファイアウォールをパススルーすることを許可します。 デフォルトでは、どの着信セッションも **conduit** や **access-list** コマンド文によって明示的に許可する必要があります。 IPSec 保護トラフィックでは、二次アクセス リストのチェックが冗長になる可能性があります。 IPSec の認証/暗号化着信セ

ッションが常に許可されるように有効にするには、**sysopt connection permit-ipsec** コマンドを使用します。

Access Control List (ACL; アクセス コントロール リスト) の確認

通常の IPSec VPN 設定ではアクセス リストを 2 つ使用します。 一方のアクセス リストは、VPN トンネルに宛てられたトラフィックを NAT プロセスから除外するために使用します。 もう一方のアクセス リストでは、暗号化するトラフィックが定義されます。 これには、LAN-to-LAN 設定のクリプト ACL、またはリモート アクセス設定のスプリット トンネリング ACL が含まれます。 これらの ACL が誤って設定されていたり、存在しなかったりすると、トラフィックが VPN トンネルを 1 方向にしか流れなかったり、トンネルにトラフィックがまったく送られなかったりします。

IPSec VPN の設定に必要なすべてのアクセス リストを設定していることと、それらのアクセス リストにトラフィックが正確に定義されていることを確認してください。 このリストには、IPSec VPN の問題の原因が ACL にあることが疑われる場合に確認する単純な項目が含まれています。

- NAT 除外 ACL とクリプト ACL でトラフィックが正しく指定されていることを確認します。
- 複数の VPN トンネルと複数のクリプト ACL がある場合は、それらの ACL が重複していないことを確認します。
- ACL を 2 回使用しないようにします。 NAT 除外 ACL とクリプト ACL で同じトラフィックが指定されている場合でも、2 つの異なるアクセス リストを使用してください。
- 使用しているデバイスで、NAT 除外 ACL を使用するように設定されていることを確認します。 つまり、ルータ上では **route-map** コマンド、 PIX または ASA 上では **nat (0)** コマンドを使用します。 NAT 除外 ACL は、LAN ツー LAN 設定とリモート アクセス設定の両方に必要です。

ACL 文を確認する方法についての詳細は、『一般的な L2L およびリモート アクセス IPSec VPN のトラブルシューティング方法について』の「ACL が正しいことを確認する」セクションを参照してください。

関連情報

- IPSec ネゴシエーション/IKE プロトコルに関するサポート ページ
- IPセキュリティ (IPSec) 暗号化入門
- PIX に関するサポート ページ
- PIX コマンド リファレンス
- Requests for Comments (RFC)
- テクニカルサポートとドキュメント - Cisco Systems