Comprendre et utiliser les commandes de débogage pour dépanner IPSec

Table des matières

Introduction

Ce document décrit les commandes de débogage courantes utilisées pour dépanner les problèmes IPsec sur le logiciel Cisco IOS® et PIX/ASA.

Conditions préalables

Exigences

Ce document part du principe que vous avez configuré IPsec. Référez-vous à Négociation IPSec/Protocoles IKE pour plus de détails.

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Logiciel Cisco IOSMD
  - Ensemble de fonctionnalités IPsec.
  - 56i : indique une Data Encryption Standard (DES) fonctionnalité unique (sur le logiciel Cisco IOS® versions 11.2 et ultérieures).
  - k2 : indique la fonctionnalité Triple DES (sur le logiciel Cisco IOS® version 12.0 et ultérieure). La triple fonctionnalité DES est disponible sur les gammes Cisco 2600 et ultérieures.
- PIX - V5.0 et ultérieures, dont l'activation requiert une clé de licence DES simple ou triple.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l’incidence possible des commandes.

Conventions

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à Conventions relatives aux conseils techniques Cisco.

Informations générales

Reportez-vous à la section Solutions de dépannage les plus courantes pour L2L et VPN IPsec à accès distant pour obtenir des informations sur les solutions les plus courantes aux problèmes liés au VPN IPsec.

Il contient une liste de contrôle des procédures courantes que vous pouvez essayer avant de commencer à dépanner une connexion et d'appeler l'assistance technique Cisco.

Débogages du logiciel Cisco IOS®

Les rubriques de cette section décrivent les commandes de débogage du logiciel Cisco IOS®. Référez-vous à Négociation IPSec/Protocoles IKE pour plus de détails.

show crypto isakmp sa

Cette commande afficheInternet Security Association Management Protocol (ISAKMP) Security Associations (SAs)la configuration établie entre les homologues.

```
dst          src        state     conn-id     slot
10.1.0.2  10.1.0.1   QM_IDLE    1           0
```

show crypto ipsec sa

Cette commande montre les SA IPsec créées entre homologues. Le tunnel crypté est créé entre 10.1.0.1 et 10.1.0.2 pour le trafic qui passe entre les réseaux 10.1.0.0 et 10.1.1.0.

Vous pouvez voir les deux Encapsulating Security Payload (ESP) SA créées en entrée et en sortie. L'AH (Authentication Header) n'est pas utilisé puisqu'il n'y a aucune SA AH.

Ce résultat montre un exemple de**show crypto ipsec sa**la commande.

```
<#root>

  interface: FastEthernet0
    Crypto map tag: test, local addr.

10.1.0.1

   local  ident (addr/mask/prot/port): (

10.1.0.0/255.255.255.0/0/0

)
```

remote ident (addr/mask/prot/port): (

10.1.1.0/255.255.255.0/0/0

)
current_peer:

10.1.0.2

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.: 10.1.0.1, remote crypto endpt.: 10.1.0.2

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

Cette commande montre chaque SA de phase 2 créée ainsi que le volume du trafic envoyé.

Comme la phase 2 Security Associations (SAs)est unidirectionnelle, chaque SA affiche le trafic dans une seule direction (les chiffrements sont sortants et les déchiffrements entrants).

debug crypto isakmp

Ce résultat montre un exemple de la **debug crypto isakmp**commande.

<#root>

```
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

Cette commande indique la source et la destination des points de terminaison du tunnel IPsec.Src_proxy et dest_proxyreprésentent les sous-réseaux clients.

sa createdDeux messages apparaissent avec un dans chaque direction. (Quatre messages apparaissent si vous exécutez ESP et AH.)

Ce résultat montre un exemple de**debug crypto ipsec**la commande.

<#root>

```
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= 10.1.0.2, src=10.1.0.1,
      dest_proxy= 10.1.1.0/0.0.0.0/0/0,
      src_proxy= 10.1.0.0/0.0.0.16/0/0,
      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 10.1.0.2 to 10.1.0.1 for prot 2
IPSEC(spi_response): getting spi 194838793 for SA
      from 10.1.0.2 to 10.1.0.1 for prot 3
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): ,
  (key eng. msg.) dest=
```

10.1.0.2

```
, src=
```

10.1.0.1

```
,
```

dest_proxy= 10.1.1.0/255.255.255.0/0/0,
**   src_proxy= 10.1.0.0/255.255.255.0/0/0,**

```
      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=
```

10.1.0.2

```
, dest=
```

```
10.1.0.1,


 src_proxy= 10.1.1.0/255.255.255.0/0/0,
      dest_proxy= 10.1.0.0/255.255.255.0/0/0,

      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= 10.1.0.2, 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= 10.1.0.2, sa_prot= 50,
      sa_spi= 0xDED0AB4(233638580),
      sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
```

Exemples de messages d'erreur

Les exemples de messages d'erreur qui suivent ont été générés par les commandes de débogage répertoriées ici :

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

Replay Check Failed

Ce résultat montre un exemple de"Replay Check Failed"l'erreur :

```
%CRYPTO-4-PKT_REPLAY_ERR: decrypt: replay check failed connection id=#.
```

Cette erreur est le résultat d'un réordre dans le support de transmission (en particulier s'il existe des chemins parallèles), ou de chemins inégaux de paquets traités dans Cisco IOS® pour les paquets de grande taille par rapport aux petits paquets plus sous charge.

Changez transform-set afin de refléter cela. Le reply check n'est visible que lorsque transform-set esp-md5-hmac est activé. Afin de supprimer ce message d'erreur, désactivezesp-md5-hmacet effectuez uniquement le chiffrement.

Référez-vous au bogue Cisco IDCSCdp19680 (clients enregistrés uniquement) .

QM FSM Error

Le tunnel VPN L2L IPsec n'apparaît pas sur le pare-feu PIX ou l'ASA et le message d'erreur QM FSM s'affiche.

Une raison possible est que les identités de proxy, telles que le trafic inhabituelAccess Control List (ACL),ou la liste de contrôle d'accès de chiffrement, ne correspondent pas aux deux extrémités.

Vérifiez la configuration des deux périphériques et assurez-vous que les ACL de chiffrement correspondent.

Une autre raison possible est une non-correspondance des paramètres du jeu de transformation. Vérifiez qu'aux deux extrémités, les passerelles VPN utilisent le même jeu de transformation avec les mêmes paramètres.

Invalid Local Address

La sortie ci-dessous est un exemple de ce message d'erreur :

```
IPSEC(validate_proposal): invalid local address 10.2.0.2
ISAKMP (0:3): atts not acceptable. Next payload is 0
ISAKMP (0:3): SA not acceptable!
```

Ce message d'erreur est attribué à l'un des deux problèmes courants suivants :

- **crypto map map-name local-address interface-id**La commande force le routeur à utiliser une adresse incorrecte comme identité, car elle force le routeur à utiliser une adresse spécifiée.
- Crypto map est appliqué à la mauvaise interface ou n'est pas appliqué du tout. Vérifiez la configuration pour vous assurer que la carte de chiffrement est bien appliquée à l'interface voulue.

IKE Message from X.X.X.X Failed its Sanity Check or is Malformed

Cette erreur de débogage se produit si les clés pré-partagées des homologues ne correspondent pas. Pour résoudre ce problème, vérifiez les clés pré-partagées des deux côtés.

```
1d00H:%CRPTO-4-IKMP_BAD_MESSAGE: IKE message from 198.51.100.1 failed its
 sanity check or is malformed
```

Échec du processus du mode principal avec l'homologue

Voici un exemple de messageMain Moded'erreur. L'échec du main mode laisse penser que la stratégie de la phase 1 ne correspond pas des deux côtés.

```
1d00h: ISAKMP (0:1): atts are not acceptable. Next payload is 0
1d00h: ISAKMP (0:1); no offers accepted!
1d00h: ISAKMP (0:1): SA not acceptable!
1d00h: %CRYPTO-6-IKMP_MODE_FAILURE: Processing of Main Mode failed with
peer at 198.51.100.1
```

Une commande show crypto isakmp sa montre que la SA ISAKMP doit être dansMM_NO_STATE. Ceci signifie également que le main mode a échoué.

```
dst          src        state              conn-id         slot
10.1.1.2  10.1.1.1   MM_NO_STATE        1           0
```

Vérifiez que la stratégie de la phase 1 se trouve bien sur les deux homologues et assurez-vous que tous les attributs correspondent.

```
Encryption DES or 3DES
Hash MD5 or SHA
Diffie-Hellman Group 1 or 2
Authentication {rsa-sig | rsa-encr | pre-share
```

Proxy Identities Not Supported

Ce message apparaît dans les débogages si la liste d'accès du trafic IPsec ne correspond pas.

```
1d00h: IPSec(validate_transform_proposal): proxy identities not supported
1d00h: ISAKMP: IPSec policy invalidated proposal
1d00h: ISAKMP (0:2): SA not acceptable!
```

Les listes d’accès de chaque homologue doivent se refléter (toutes les entrées doivent être réversibles). L'exemple ci-dessous illustre ce point.

```
Peer A
```

```
access-list 150 permit ip 172.21.113.0 0.0.0.255 172.21.114.0 0.0.0.255
access-list 150 permit ip host 10.2.0.8 host 172.21.114.123
Peer B
access-list 150 permit ip 172.21.114.0 0.0.0.255 172.21.113.0 0.0.0.255
access-list 150 permit ip host 172.21.114.123 host 10.2.0.8
```

Transform Proposal Not Supported

Ce message s'affiche si la phase 2 (IPsec) ne correspond pas des deux côtés. Ceci se produit le plus souvent en cas de non correspondance ou d'incompatibilité dans le jeu de transformations.

```
1d00h: IPSec (validate_proposal): transform proposal
   (port 3, trans 2, hmac_alg 2) not supported
1d00h: ISAKMP (0:2) : atts not acceptable. Next payload is 0
1d00h: ISAKMP (0:2) SA not acceptable
```

Vérifiez que le jeu de transformations correspond des deux côtés :

```
crypto ipsec transform-set transform-set-name transform1
[transform2 [transform3]]
? ah-md5-hmac
? ah-sha-hmac
? esp-des
? esp-des and esp-md5-hmac
? esp-des and esp-sha-hmac
? esp-3des and esp-md5-hmac
? esp-3des and esp-sha-hmac
? comp-lzs
```

No Cert and No Keys with Remote Peer

Ce message indique que l'adresse de l'homologue configurée sur le routeur n'est pas valide ou qu'elle a changé. Vérifiez que l'adresse de l'homologue est correcte et qu'elle peut être atteinte.

```
1d00h: ISAKMP: No cert, and no keys (public or pre-shared) with
remote peer 198.51.100.2
```

Peer Address X.X.X.X Not Found

Ce message d'erreur apparaît normalement avecVPN 3000 Concentrator le message"Message: No proposal chosen(14)"d'erreur. En effet, les connexions sont de type hôte à hôte.

D'après l'ordre d'apparition des propositions IPsec dans la configuration du routeur, la proposition choisie pour le routeur correspond à la liste d'accès mais pas à l'homologue.

La liste d'accès dispose d'un réseau plus vaste, incluant l'hôte qui croise le trafic. Pour corriger ce problème, faites passer la proposition du routeur pour cette connexion concentrateur-routeur en premier.

Elle sera ainsi d'abord mise en correspondance avec l'hôte spécifique.

```
20:44:44: IPSEC(validate_proposal_request): proposal part #1,
  (key eng. msg.) dest= 192.0.2.15, src=198.51.100.6,
    dest_proxy= 10.0.0.76/255.255.255.255/0/0 (type=1),
    src_proxy= 198.51.100.23/255.255.255.255/0/0 (type=1),
    protocol= ESP, transform= esp-3des esp-md5-hmac ,
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
20:44:44: IPSEC(validate_transform_proposal):
   peer address 198.51.100.6 not found
```

IPsec Packet has Invalid SPI

La sortie ci-dessous est un exemple de ce message d'erreur :

```
%PIX|ASA-4-402101: decaps: recd IPSEC packet has
invalid spi for destaddr=dest_address, prot=protocol, spi=number
```

Le paquet IPsec reçu spécifie un Security Parameters Index (SPI) qui n'existe pas dans le Security Associations Database (SADB). Il peut s'agir d'un problème temporaire, dû à :

- Légères différences dans l'ancienneté des Security Sssociations (SAs) homologues IPsec.
- Les SA locales ont été effacées.
- Paquets incorrects envoyés par l'homologue IPsec.

C'est peut-être une attaque.

Action recommandée :

L'homologue ne reconnaît peut-être pas que les associations de sécurité locales ont été effacées. Si une nouvelle connexion est établie à partir du routeur local, les deux homologues peuvent se reconnecter. Sinon, si le problème se produit pendant plus d'une courte période, essayez d'établir une nouvelle connexion ou contactez l'administrateur de cet homologue.

PSEC(initialize_sas) : ID de proxy non valides

L'erreur "21:57:57: IPSEC(initialize_sas): invalid proxy IDs"indique que l'identité de proxy reçue ne correspond pas à l'identité de proxy configurée selon la liste d'accès.

Pour vous assurer qu'elles correspondent toutes les deux, vérifiez la sortie de la commande de débogage.

Dans le résultat de la commande debug de la demande de proposition, access-list 103 permit ip 10.1.1.0 0.0.0.255 10.1.0.0 0.0.0.255 ne correspond pas.

La liste d'accès est spécifique au réseau à une extrémité et à l'hôte à l'autre.

```
21:57:57: IPSEC(validate_proposal_request): proposal part #1,
  (key eng. msg.) dest= 192.0.2.1, src=192.0.2.2,
    dest_proxy= 10.1.1.1/255.255.255.0/0/0 (type=4),
    src_proxy= 10.2.0.1/255.255.255.0/0/0 (type=4)
```

Reserved Not Zero on Payload 5

Cette commande signifie que les clés ISAKMP ne correspondent pas. Procédez à une nouvelle saisie/réinitialisation afin de garantir l'exactitude.

Hash Algorithm Offered does not Match Policy

Si les stratégies ISAKMP configurées ne correspondent pas à la stratégie proposée par l'homologue distant, le routeur essaye la stratégie par défaut 65535.

Si elle ne correspond pas non plus, la négociation ISAKMP échoue.

Un utilisateur reçoit l'un ou l'autre"Hash algorithm offered does not match policy!""Encryption algorithm offered does not match policy!"des messages d'erreur sur les routeurs.

<#root>

```
=RouterA=
3d01h: ISAKMP (0:1): processing SA payload. message ID = 0
3d01h: ISAKMP (0:1): found peer pre-shared key matched 203.0.113.22
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

Ce message d'erreur est signalé en cas d'échec de la vérification deHash Message Authentication Codela sur le paquet IPsec. Ceci se produit généralement lorsque le paquet est corrompu.

<#root>

```
Sep 22 11:02:39 203.0.113.16 2435:
Sep 22 11:02:39:

%MOTCR-1-ERROR:motcr_crypto_callback() motcr return failure

Sep 22 11:02:39 203.0.113.16 2436:
Sep 22 11:02:39:

%MOTCR-1-PKTENGRET_ERROR: MOTCR PktEng Return Value = 0x20000,
                 PktEngReturn_MACMiscompare
```

Si vous rencontrez occasionnellement ce message d'erreur, vous pouvez l'ignorer. Cependant, si cela devient plus fréquent, vous devez alors enquêter sur la source de la corruption du paquet. Il peut s'agir d'un défaut au niveau de l'accélérateur de chiffrement.

Remote Peer Not Responding

Ce message d'erreur s'affiche en cas de non correspondance au niveau d'un jeu de transformations. Assurez-vous que les jeux de transformation correspondants sont configurés sur

les deux homologues.

Toutes les propositions IPSec SA jugées inacceptables

Ce message d'erreur se produit lorsque les paramètres IPSec de phase 2 ne correspondent pas entre les sites local et distant.

Afin de résoudre ce problème, spécifiez les mêmes paramètres dans le jeu de transformation afin qu'ils correspondent et que le VPN réussisse à établir.

Packet Encryption/Decryption Error

La sortie ci-dessous est un exemple de ce message d'erreur :

```
HW_VPN-1-HPRXERR: Virtual Private Network (VPN) Module0/2: Packet Encryption/Decryption
error, status=4615
```

Ce message d'erreur est probablement dû à l'une des raisons suivantes :

- Fragmentation - Les paquets de chiffrement fragmentés sont commutés par processus, ce qui force l'envoi des paquets à commutation rapide vers la carte VPN, avant les paquets commutés par processus.

Si un nombre suffisant de paquets à commutation rapide sont traités avant les paquets commutés par processus, le numéro de séquence de l'ESP ou de l'AH pour le paquet commuté par processus devient obsolète et lorsque le paquet arrive au niveau de la carte VPN, son numéro de séquence se trouve en dehors de la fenêtre de relecture.

Ceci provoque des erreurs de numéro de séquence de l'AH ou de l'ESP (4615 et 4612, respectivement), selon l'encapsulation que vous utilisez.

- Entrées de cache obsolètes - Ceci peut également se produire lorsqu'une entrée de cache à commutation rapide devient obsolète et que le premier paquet avec un élément non retrouvé en cache est commuté par processus.

Solution De Contournement

1. Désactivez toute authentification dans le jeu de transformations 3DES et utilisez l'ESP-DES/3DES. Cela désactive efficacement la protection d'authentification/anti-relecture, qui (à son tour) empêche les erreurs d'abandon de paquets liées au trafic IPsec non ordonné (mixte)%HW_VPN-1-HPRXERR: Hardware VPN0/2: Packet Encryption/Decryption error, status=4615.
2. Une solution de contournement qui s'applique à la raison mentionnée ici est de définir la taille Maximum Transmission Unit (MTU) des flux entrants à moins de 1400 octets. Pour ce faire, entrez la commande suivante :

```
ip tcp adjust-mss 1300
```

3. Désactivez la carte AIM.
4. Désactivez la commutation rapide/CEF sur les interfaces du routeur. Afin de supprimer la commutation rapide, utilisez cette commande en mode de configuration d'interface :

```
no ip route-cache
```

Packets Receive Error Due to ESP Sequence Fail

Voici un exemple du message d'erreur :

```
%C1700_EM-1-ERROR: packet-rx error: ESP sequence fail
```

Ce message d'erreur indique généralement l'un des problèmes suivants :

- Les paquets IPsec cryptés sont transférés dans n'importe quel ordre par le routeur de cryptage en raison d'un mécanisme QoS mal configuré.
- Les paquets IPsec reçus par le routeur de décryptage sont dans le désordre en raison d'un réordre de paquets au niveau d'un périphérique intermédiaire.
- Le paquet IPsec reçu est fragmenté et nécessite un réassemblage avant de pouvoir procéder au déchiffrement et à la vérification de l'authentification.

Solution de contournement

1. Désactivez QoS pour le trafic IPsec sur les routeurs de cryptage ou intermédiaires.
2. Activez la pré-fragmentation IPsec sur le routeur de cryptage.

```
<#root>
Router(config-if)#
crypto ipsec fragmentation before-encryption
```

3. Définissez la valeur de MTU sur une taille qui ne nécessite pas de fragmentation.

```
<#root>

Router(config)#

interface type [slot_#/]port_#
```

```
<#root>

Router(config-if)#

ip mtu MTU_size_in_bytes
```

4. Mettez à niveau l'image Cisco IOS® vers la dernière image stable disponible dans cette catégorie.

Si la taille de MTU est modifiée sur un routeur, tous les tunnels terminés sur cette interface doivent être démantelés.

Prévoyez de compléter cette solution de contournement pendant une période d'indisponibilité planifiée.

Error Trying to Establish VPN Tunnel on 7600 Series Router

Cette erreur se produit lorsque vous tentez d'établir un tunnel VPN sur des routeurs de la gamme 7600 :

```
crypto_engine_select_crypto_engine: can't handle any more
```

Cette erreur se produit parce que le chiffrement logiciel n'est pas pris en charge sur les routeurs de la gamme 7600. Les routeurs de la gamme 7600 ne prennent pas en charge l'arrêt des tunnels IPsec sans matériel IPsec SPA. Le VPN est uniquement pris en charge avec une carte IPSEC-SPA dans les routeurs 7600.

Débogages PIX

show crypto isakmp sa

Cette commande montre la SA ISAKMP créée entre homologues.

```
dst          src        state     conn-id     slot
10.1.0.2  10.1.0.1   QM_IDLE    1           0
```

Dans le show crypto isakmp sa output, l'état doit toujours être `QM_IDLE`. Si l'état est MM_KEY_EXCH, cela signifie soit que la clé pré-partagée configurée est incorrecte, soit que les adresses IP diffèrent.

<#root>

```
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
```

Vous pouvez rectifier ceci lorsque vous configurez l'adresse IP ou la clé pré-partagée correcte.

show crypto ipsec sa

Cette commande montre les SA IPsec créées entre homologues. Un tunnel crypté est créé entre 10.1.0.1 et 10.1.0.2 pour le trafic qui passe entre les réseaux 10.1.0.0 et 10.1.1.0.

Vous pouvez voir les deux SA ESP créées, en entrée et en sortie. L'AH n'est pas utilisé puisqu'il n'y a aucune SA AH.

Un exemple de**show crypto ipsec sa**la commande est présenté dans ce résultat.

<#root>

```
interface: outside
    Crypto map tag: vpn, local addr. 10.1.0.1
   local  ident (addr/mask/prot/port): (
```

10.1.0.0/255.255.255.0/0/0

```
)
   remote ident (addr/mask/prot/port): (
```

10.1.0.2/255.255.255.255/0/0

```
)
   current_peer: 10.2.1.1
```

dynamic allocated peer ip: 10.1.0.2

```
     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.: 10.1.0.1, remote crypto endpt.: 10.1.0.2
     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

Cette commande affiche des informations de débogage sur les connexions IPsec et indique le premier ensemble d'attributs refusés en raison d'incompatibilités aux deux extrémités.

La deuxième tentative de correspondance (pour essayer 3DES au lieu de DES) etSecure Hash Algorithm (SHA)c'est acceptable, et la SA ISAKMP est construite.

Ce débogage provient également d'un client à distance acceptant une adresse IP (10.32.8.1) qui ne fait pas partie d'un pool local. Une fois la SA ISAKMP créée, les attributs IPsec sont négociés et déclarés acceptables.

Le PIX configure alors les SA IPsec comme indiqué ci-dessous. Ce résultat montre un exemple de**debug crypto isakmp**la commande.

<#root>

```
crypto_isakmp_process_block: src 10.1.0.1, dest 10.1.0.2
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 10.1.0.2
OAK_QM exchange
ISAKMP (0:0): Need config/address
ISAKMP (0:0): initiating peer config to 10.1.0.2. ID = 2607270170 (0x9b67c91a)
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 10.1.0.2, dest 10.1.0.1
ISAKMP_TRANSACTION exchange
ISAKMP (0:0): processing transaction payload from 10.1.0.2.
   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 10.1.0.1 prot 0 port 0
INITIAL_CONTACTIPSEC(key_engine): got a queue event...
```

debug crypto ipsec

Cette commande affiche des informations de débogage sur les connexions IPsec.

<#root>

```
IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 0xd532efbd(3576885181) for SA
        from  10.1.0.2  to  10.1.0.1  for prot 3
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 10.1.0.2, dest 10.1.0.1
OAK_QM exchange
oakley_process_quick_mode:
OAK_QM_AUTH_AWAIT
ISAKMP (0):

Creating IPSec SAs
        inbound SA from  10.1.0.2  to  10.1.0.1
           (proxy 10.32.8.1 to  10.1.0.1.)
        has spi 3576885181 and conn_id 2 and flags 4
        outbound SA from  10.1.0.1  to  10.1.0.2
           (proxy 10.1.0.1 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= 10.1.0.1, src=10.1.0.2,
    dest_proxy= 10.1.0.1/0.0.0.0/0/0 (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=10.1.0.1, dest= 10.1.0.2,
    src_proxy= 10.1.0.1/0.0.0.0/0/0 (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
```

Problèmes courants entre routeur et client VPN

Impossible d'accéder aux sous-réseaux en dehors du tunnel VPN : tunnel partagé

Cet exemple de sortie de configuration de routeur montre comment activer un tunnel partagé pour

les connexions VPN.

La **split tunnel**commande est associée au groupe tel que configuré dans**crypto isakmp client configuration group hw-client-groupname**la commande.

Cela permet à l'Cisco VPN Client d'utiliser le routeur afin d'accéder à un sous-réseau supplémentaire qui ne fait pas partie du tunnel VPN.

Cela se fait sans compromettre la sécurité de la connexion IPsec. Le tunnel est formé sur le réseau 192.0.2.18.

Le trafic circule sans être chiffré vers des périphériques non définis dans**access list 150**la commande, tels qu'Internet.

<#root>

```
!

crypto isakmp client configuration group hw-client-groupname

 key hw-client-password
 dns 192.0.2.20 198.51.100.21
 wins 192.0.2.22 192.0.2.23
 domain cisco.com
 pool dynpool


acl 150


!
!

access-list 150 permit ip 192.0.2.18 0.0.0.127 any

!
```

Problèmes courants entre Pix et client VPN

Les sujets de cette section traitent des problèmes fréquemment rencontrés lors de la configuration de PIX vers IPsec à l'aide du client VPN 3 x. Les exemples de configuration pour le PIX sont basés sur la version 6.x.

Le trafic ne circule pas après l'établissement du tunnel : impossible d'envoyer une requête ping à l'intérieur du réseau derrière PIX

Il s'agit d'un problème courant, relatif au routage. Assurez-vous que le PIX dispose d'une route pour les réseaux qui se trouvent à l'intérieur et ne sont pas directement connectés au même sous-réseau.

Le réseau interne doit également disposer d'une route de retour au PIX pour les adresses du pool

d'adresses du client.

La sortie ci-dessous est un exemple.

```
!--- Address of PIX inside interface.

ip address inside 10.1.1.1 255.255.255.240

!--- Route to the networks that are on the inside segment. !--- The next hop is the router on the inside

route inside 172.16.0.0 255.255.0.0 10.1.1.2 1

 !--- Pool of addresses defined on PIX from which it assigns !--- addresses to the VPN Client for the I

ip local pool mypool 10.1.2.1-10.1.2.254

!--- On the internal router, if the default gateway is not !--- the PIX inside interface, then the route

ip route 10.1.2.0 255.255.255.0 10.1.1.1
```

Une fois le tunnel activé, l'utilisateur ne peut pas naviguer sur Internet : Split Tunnel

Ce problème provient généralement du fait qu'avec le tunnel IPsec du client VPN vers le PIX, tout le trafic est envoyé par le tunnel vers le pare-feu PIX.

La fonctionnalité PIX ne permet pas le renvoi du trafic vers l'interface sur laquelle il a été reçu. Par conséquent, le trafic destiné à Internet ne fonctionne pas.

Afin de résoudre ce problème, utilisez**split tunnel**la commande. Cette résolution permet d'envoyer uniquement un trafic spécifique par le tunnel, et le reste directement sur Internet, et non par le tunnel.

<#root>

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
```

vpngroup vpn3000 split-tunnel 90 La commande active le split tunnel avecaccess-list number 90.

access-list number 90 La commande définit le trafic qui traverse le tunnel, dont le reste est refusé à la fin de la liste d'accès.

La liste d'accès doit être la même pour êtreNetwork Address Translation (NAT)refusée sur PIX.

Une fois le tunnel activé, certaines applications ne fonctionnent pas : réglage MTU

sur le client

Une fois le tunnel établi, bien que vous puissiez envoyer une requête ping aux machines sur le réseau derrière le pare-feu PIX, vous ne pouvez pas utiliser certaines applications comme Microsoft

Perspectives

La taille de MTU (Maximum Transfer Unit) des paquets est un problème courant. L'en-tête IPsec peut faire jusqu'à 50 à 60 octets, qui sont ajoutés au paquet d'origine.

Si la taille du paquet est supérieure à 1 500 (valeur par défaut pour Internet), les périphériques doivent procéder à une fragmentation. Une fois l'en-tête IPsec ajoutée, la taille est toujours inférieure à 1 496, ce qui est la valeur maximale pour IPsec.

La **show interface**commande affiche le MTU de cette interface particulière sur les routeurs qui sont accessibles ou sur les routeurs dans vos propres locaux.

Afin de déterminer le MTU du chemin entier de la source à la destination, les datagrammes de différentes tailles sont envoyés avecDo Not Fragment (DF)le bit défini de sorte que, si le datagramme envoyé est supérieur au MTU, ce message d'erreur est renvoyé à la source :

```
frag. needed and DF set
```

La sortie ci-dessous est un exemple de recherche de la MTU du chemin entre les hôtes dont les adresses IP sont 10.1.1.2 et 172.16.1.56.

<#root>

```
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
```

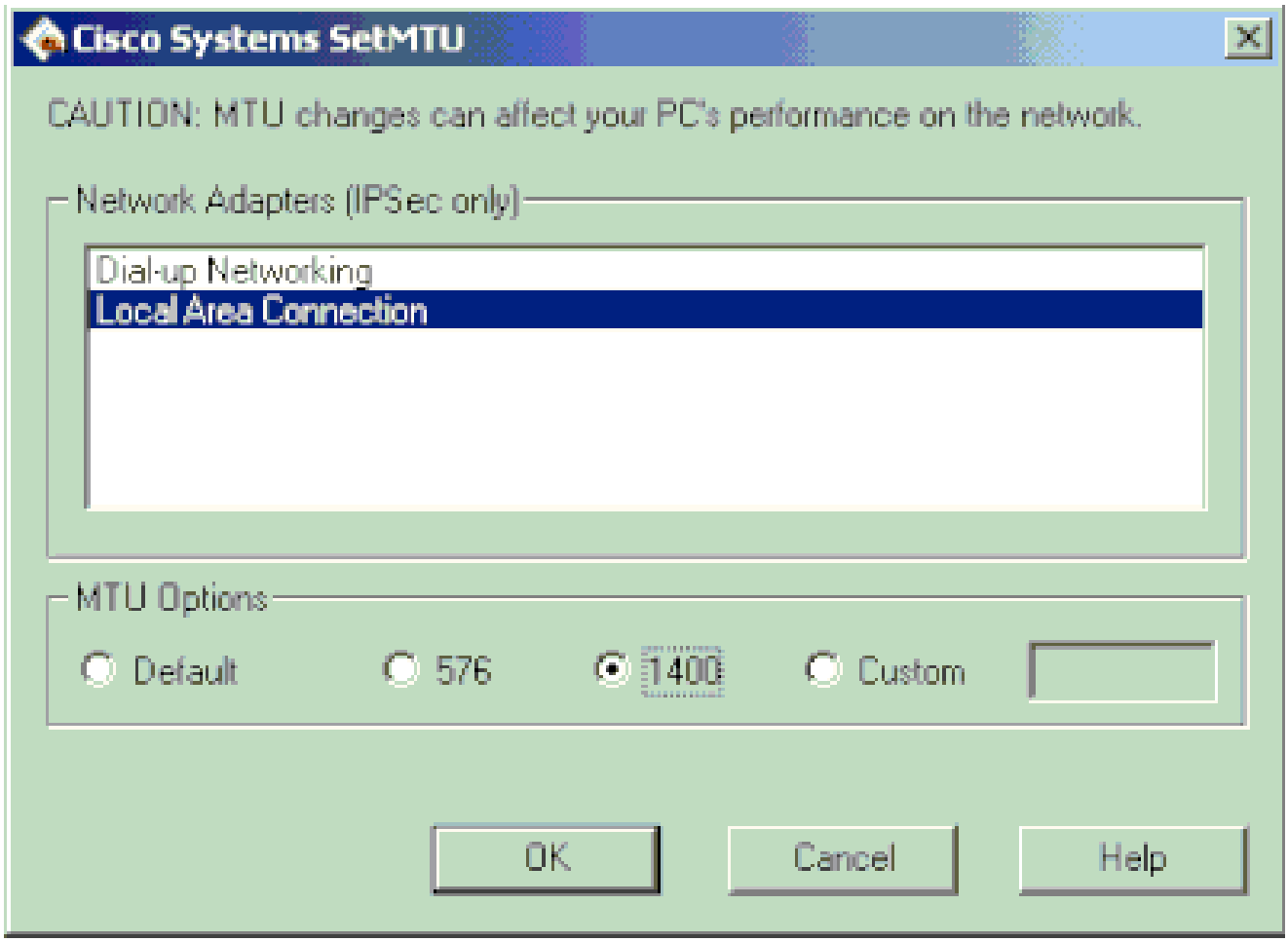
```
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 172.16.1.56, timeout is 2 seconds:
!!!!!
2w5d: ICMP: echo reply rcvd, src 172.16.1.56, dst 10.1.1.2
2w5d: 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
```

le client VPN est livré avec un utilitaire de réglage de la MTU qui permet à l'utilisateur de régler la MTU pour le client VPN Cisco.

Pour les utilisateurs du client PPP over Ethernet (PPPoE), réglez la MTU pour l'adaptateur PPPoE.

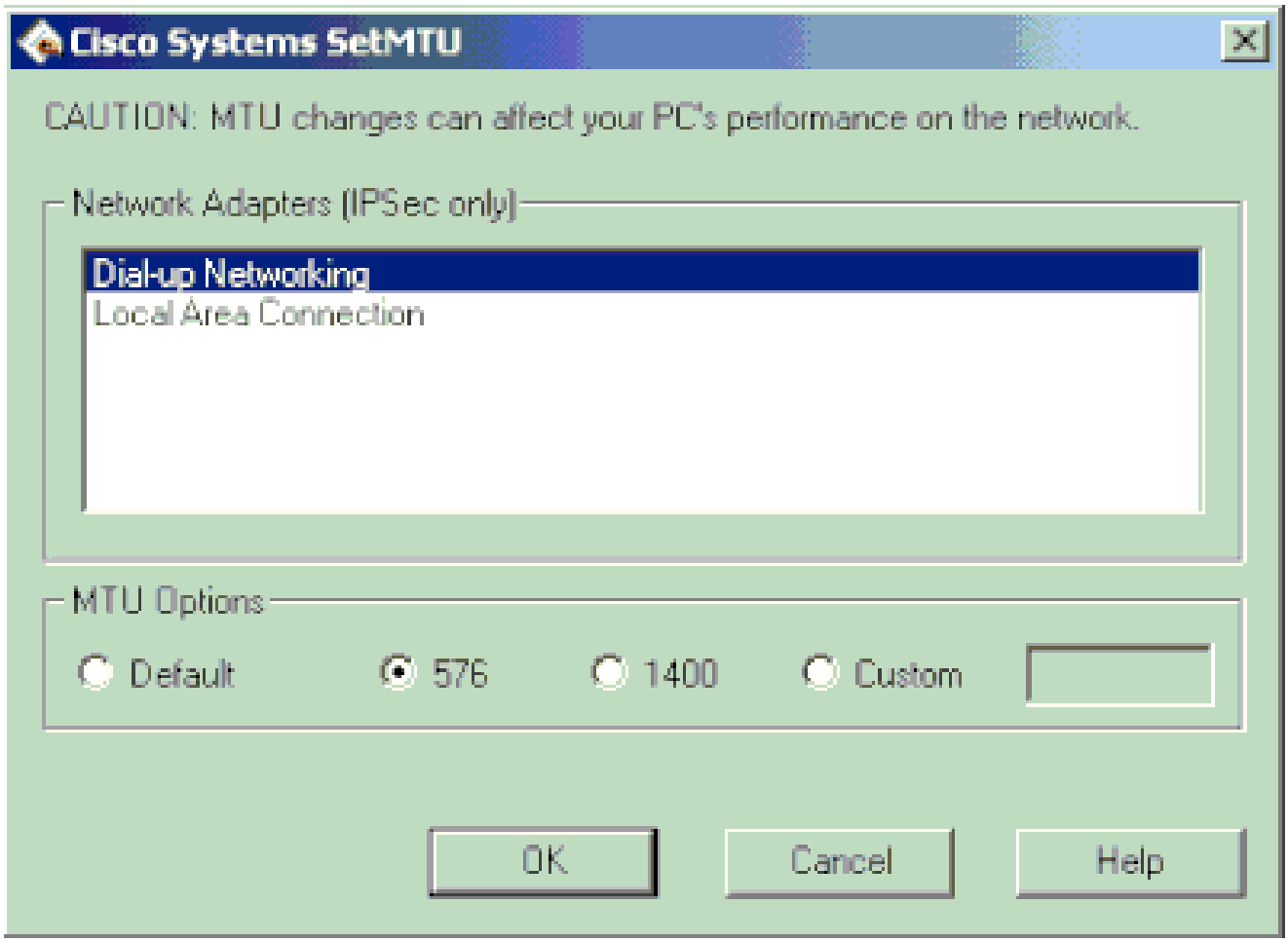
pour régler l'utilitaire MTU pour le client VPN, procédez comme suit :

1. Choisir **Start** > **Programs** > **Cisco System VPN Client** > **Set MTU**.
2. Sélectionnez**Local Area Connection**, puis cliquez sur la case d'option 1400.
3. Cliquer**OK**.



4. Répétez l'étape 1 et sélectionnez**Dial-up Networking**.

5. Cliquez sur le576puis cliquez sur l'option**OK**.



Commande sysopt manquée

Utilisez la **sysopt connection permit-ipsec**commande dans les configurations IPsec sur le PIX afin d'autoriser le trafic IPsec à passer à travers le pare-feu PIX sans une vérification des**conduit**ou**access-list**instructions ou de commande.

Par défaut, toute session entrante doit être explicitement autorisée par une instruction **conduit** ou **access-list**command. Le trafic étant protégé par IPsec, il se peut que la vérification secondaire de la liste d'accès soit redondante.

Afin d'activer l'autorisation permanente des sessions entrantes IPsec authentifiées/chiffrées, utilisez**sysopt connection permit-ipsec**la commande.

Vérification des listes de contrôle d'accès (ACL)

Dans une configuration VPN IPsec classique, deux listes d'accès sont utilisées. L'une permet d'exempter le trafic destiné au tunnel VPN à partir du processus NAT,

l'autre définit le trafic à crypter. Cela inclut une ACL de chiffrement dans une configuration de LAN à LAN ou une ACL de tunnel partagé dans une configuration d'accès à distance.

Lorsque ces listes de contrôle d'accès sont mal configurées ou manquées, le trafic circule peut-être dans une seule direction à travers le tunnel VPN, ou il n'a pas été envoyé à travers le tunnel.

Assurez-vous d'avoir configuré toutes les listes d'accès nécessaires pour réaliser votre configuration VPN IPsec et que ces listes d'accès définissent le trafic voulu.

Cette liste contient les éléments à vérifier lorsque vous suspectez qu'une ACL est à l'origine des problèmes que vous rencontrez avec votre VPN IPsec.

- Assurez-vous que votre exemption NAT et vos listes de contrôle d'accès de chiffrement spécifient le trafic voulu.
- Si vous avez plusieurs tunnels VPN et ACL de chiffrement, assurez-vous que ces ACL ne se superposent pas.
- N'utilisez pas une ACL deux fois. Même si vos listes de contrôle d'accès d'exemption NAT et de chiffrement indiquent le même trafic, utilisez deux listes d'accès différentes.
- Assurez-vous que votre périphérique est configuré pour utiliser la liste de contrôle d'accès d'exemption NAT : Autrement dit, utilisez**route-map**la commande sur le routeur ; utilisez**nat (0)**la commande sur le PIX ou l'ASA. Une liste de contrôle d'accès d'exemption NAT est requise pour les configurations de LAN-à-LAN et les configurations d'accès à distance.

Afin d'en savoir plus sur la façon de vérifier les instructions de liste de contrôle d'accès, référez-vous à la sectionVérifier que les listes de contrôle d'accès sont correctes dansSolutions de dépannage VPN IPsec L2L et d'accès à distance les plus courantes.

Informations connexes

- Page de support pour Protocole IKE/Négociation Ipsec
- Page de support PIX
- Notes techniques
- Assistance et documentation techniques - Cisco Systems

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.