

# strongSwan en tant que client vpn d'Accès à distance (Xauth) qui se connecte à l'exemple de configuration du logiciel de Cisco IOS

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Configurez](#)

[Topologie](#)

[Configurez le logiciel de Cisco IOS](#)

[Configurez strongSwan](#)

[Vérifiez](#)

[Dépannez](#)

[Résumé](#)

[Informations connexes](#)

## Introduction

Ce document décrit comment configurer strongSwan en tant que client vpn d'IPSec d'Accès à distance qui connecte au Cisco IOS® le logiciel.

strongSwan est le logiciel libre qui est utilisé afin d'établir des tunnels VPN de l'Échange de clés Internet (IKE) /IPSec et construire des tunnels d'entre réseaux locaux et d'Accès à distance avec le logiciel de Cisco IOS.

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Configuration de Linux
- Configuration du VPN sur le logiciel de Cisco IOS

[Composants utilisés](#)

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

- Version du logiciel Cisco IOS 15.3T
- 5.0.4 strongSwan
- Kernel Linux 3.2.12

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

## Configurez

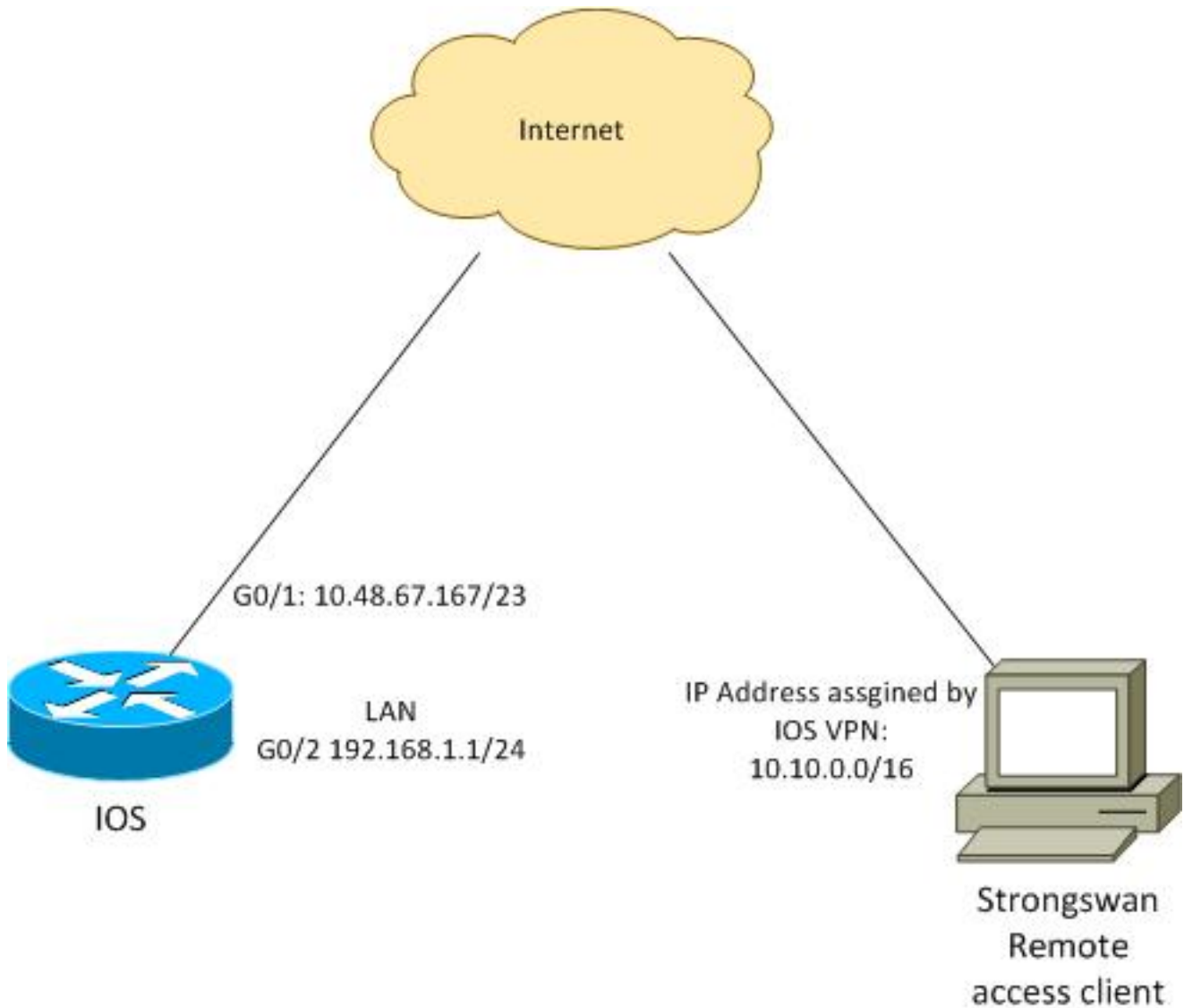
Remarques :

Utilisez l'[Outil de recherche de commande](#) (clients [enregistrés](#) seulement) pour obtenir plus d'informations sur les commandes utilisées dans cette section.

[L'Output Interpreter Tool](#) (clients [enregistrés](#) seulement) prend en charge certaines **commandes show**. Utilisez l'Output Interpreter Tool afin de visualiser une analyse de sortie de commande show.

Référez-vous aux [informations importantes sur les commandes de débogage](#) avant d'utiliser les commandes de **débogage**.

## Topologie



Le client distant reçoit une adresse IP du groupe 10.10.0.0/16. Le trafic entre 10.10.0.0/16 et 192.168.1.0/24 est protégé.

## Configurez le logiciel de Cisco IOS

Dans cet exemple, le client strongSwan a besoin de l'accès sécurisé au réseau 192.168.1.0/24 de RÉSEAU LOCAL de logiciel de Cisco IOS. Le client distant utilise le nom de groupe du RA (c'est l'IKEID) aussi bien que le nom d'utilisateur de Cisco et de mot de passe de Cisco.

Le client obtient l'adresse IP du groupe 10.10.0.0/16. En outre, la liste de contrôle d'accès fendue (ACL) est poussée au client ; cet ACL forcera le client pour envoyer le trafic à 192.168.1.0/24 par l'intermédiaire du VPN.

```
aaa new-model
aaa authentication login AUTH local
aaa authorization network NET local
username cisco password 0 cisco
```

```
crypto isakmp policy 1
 encryption aes
 hash sha
 authentication pre-share
```

```

group 2
lifetime 3600
crypto isakmp keepalive 10

crypto isakmp client configuration group RA
key cisco
domain cisco.com
pool POOL
acl split
save-password
netmask 255.255.255.0

crypto isakmp profile test
match identity group RA
client authentication list AUTH
isakmp authorization list NET
client configuration address respond
client configuration group RA
virtual-template 1

crypto ipsec transform-set test esp-aes esp-sha-hmac
mode tunnel

crypto ipsec profile ipsecprof
set security-association lifetime kilobytes disable
set transform-set test
set isakmp-profile test

interface GigabitEthernet0/1
ip address 10.48.67.167 255.255.254.0
!
interface GigabitEthernet0/2
description LAN
ip address 192.168.1.1 255.255.255.0

interface Virtual-Templatel type tunnel
ip unnumbered GigabitEthernet0/1
tunnel source GigabitEthernet0/1
tunnel mode ipsec ipv4
tunnel protection ipsec profile ipsecprof

ip local pool POOL 10.10.0.0 10.10.255.255
ip access-list extended split
permit ip host 192.168.1.1 any

```

Cisco recommande que vous n'assigniez pas l'adresse IP statique habituelle sur un virtual-template. Les interfaces d'accès virtuel sont copiées et héritent de leur configuration du virtual-template de parent, qui pourrait créer les adresses IP en double. Cependant, le virtual-template se rapporte à une adresse IP par le mot clé de « ip unnumbered » afin de remplir table de juxtaposition. Le mot clé de « ip unnumbered » est juste une référence à une adresse IP physique ou logique sur le routeur.

Pour la compatibilité en avant avec le routage d'IKE dans IKEv2, utilisez une adresse intérieure, et évitez l'utilisation adresses locales d'IPSec les « en tant que « ip unnumbered. »

## Configurez strongSwan

Cette procédure décrit comment configurer strongSwan :

1. Utilisez cette configuration dans le fichier de /etc/ipsec.conf :

```

version 2
config setup
    strictcrlpolicy=no
    charondebug="ike 4, knl 4, cfg 2" #useful debugs

conn %default
    ikelifetime=1440m
    keylife=60m
    rekeymargin=3m
    keyingtries=1
    keyexchange=ikev1
    authby=xauthpsk

conn "ezvpn"
    keyexchange=ikev1
    ikelifetime=1440m
    keylife=60m
    aggressive=yes
    ike=aes-sha1-modp1024 #Phase1 parameters
    esp=aes-sha1 #Phase2 parameters
    xauth=client #Xauth client mode
    left=10.48.62.178 #local IP used to connect to IOS
    leftid=RA #IKEID (group name) used for IOS
    leftsourceip=%config #apply received IP
    leftauth=psk
    rightauth=psk
    leftauth2=xauth #use PSK for group RA and Xauth for user cisco
    right=10.48.67.167 #gateway (IOS) IP
    rightsubnet=192.168.1.0/24
    xauth_identity=cisco #identity for Xauth, password in ipsec.secrets
    auto=add

```

Le mot clé de rightsubnet a été placé afin d'indiquer ce que le trafic devrait être protégé. Dans ce scénario, l'association de sécurité d'IPSec (SA) est établie entre 192.168.1.0/24 (sur le logiciel de Cisco IOS) et l'adresse IP strongSwan, qui est reçue du groupe 10.10.0.0/16.

Sans rightsubnet spécifié, vous pourriez compter avoir le réseau de 0.0.0.0 et IPSec SA entre l'adresse IP de client et le réseau de 0.0.0.0. C'est le comportement quand le logiciel de Cisco IOS est utilisé en tant que client.

Mais cette attente n'est pas correcte pour strongSwan. Sans rightsubnet défini, strongSwan propose une adresse IP de passerelle externe (logiciel de Cisco IOS) dans phase2 de la négociation ; dans ce scénario, cette passerelle est 10.48.67.167. Puisque le but est de protéger le trafic qui va à un RÉSEAU LOCAL interne sur le logiciel de Cisco IOS (192.168.1.0/24) et pas à une adresse IP externe de logiciel de Cisco IOS, le rightsubnet a été utilisé.

2. Utilisez cette configuration dans le fichier de /etc/ipsec.secrets :

```

10.48.67.167 : PSK "cisco" #this is PSK for group password
cisco : XAUTH "cisco" #this is password for XAuth (user cisco)

```

**Vérifiez**

Utilisez cette section pour confirmer que votre configuration fonctionne correctement.

Cette procédure décrit comment tester et vérifier la configuration strongSwan :

## 1. Le début strongSwan avec met au point activé :

```
gentool ~ # /etc/init.d/ipsec start
* Starting ...
Starting strongSwan 5.0.4 IPsec [starter]...
Loading config setup
  strictcrpolicym=no
  charondebug=ike 4, knl 4, cfg 2
Loading conn %default
  ikelifetime=1440m
  keylife=60m
  rekeymargin=3m
  keyingtries=1
  keyexchange=ikev1
  authby=xauthpsk
Loading conn 'ezvpn'
  keyexchange=ikev1
  ikelifetime=1440m
  keylife=60m
  aggressive=yes
  ike=aes-shal-modp1024
  esp=aes-shal
  xauth=client
  left=10.48.62.178
  leftid=RA
  leftsourceip=%config
  leftauth=psk
  rightauth=psk
  leftauth2=xauth
  right=10.48.67.167
  rightsubnet=192.168.1.0/24
  xauth_identity=cisco
  auto=add
found netkey IPsec stack
No leaks detected, 9 suppressed by whitelist
```

## 2. Quand le tunnel de strongSwan est initié, toutes les informations générales sur phase1, Xauth, et phase2 sont affichées :

```
gentool ~ # ipsec up ezvpn
initiating Aggressive Mode IKE_SA ezvpn[1] to 10.48.67.167
generating AGGRESSIVE request 0 [ SA KE No ID V V V V ]
sending packet: from 10.48.62.178[500] to 10.48.67.167[500] (374 bytes)
received packet: from 10.48.67.167[500] to 10.48.62.178[500] (404 bytes)
parsed AGGRESSIVE response 0 [ SA V V V V V KE ID No HASH NAT-D NAT-D ]
received Cisco Unity vendor ID
received DPD vendor ID
received unknown vendor ID: 8d:75:b5:f8:ba:45:4c:6b:02:ac:bb:09:84:13:32:3b
received XAuth vendor ID
received NAT-T (RFC 3947) vendor ID
generating AGGRESSIVE request 0 [ NAT-D NAT-D HASH ]
sending packet: from 10.48.62.178[500] to 10.48.67.167[500] (92 bytes)
received packet: from 10.48.67.167[500] to 10.48.62.178[500] (92 bytes)
parsed INFORMATIONAL_V1 request 3265561043 [ HASH N((24576)) ]
received (24576) notify
received packet: from 10.48.67.167[500] to 10.48.62.178[500] (68 bytes)
```

```

parsed TRANSACTION request 4105447864 [ HASH CP ]
generating TRANSACTION response 4105447864 [ HASH CP ]
sending packet: from 10.48.62.178[500] to 10.48.67.167[500] (76 bytes)
received packet: from 10.48.67.167[500] to 10.48.62.178[500] (68 bytes)
parsed TRANSACTION request 1681157416 [ HASH CP ]
XAuth authentication of 'cisco' (myself) successful
IKE_SA ezvpn[1] established between 10.48.62.178[RA]...10.48.67.167[10.48.67.167]
scheduling reauthentication in 86210s
maximum IKE_SA lifetime 86390s
generating TRANSACTION response 1681157416 [ HASH CP ]
sending packet: from 10.48.62.178[500] to 10.48.67.167[500] (68 bytes)
generating TRANSACTION request 1406391467 [ HASH CP ]
sending packet: from 10.48.62.178[500] to 10.48.67.167[500] (68 bytes)
received packet: from 10.48.67.167[500] to 10.48.62.178[500] (68 bytes)
parsed TRANSACTION response 1406391467 [ HASH CP ]
installing new virtual IP 10.10.0.1
generating QUICK_MODE request 1397274205 [ HASH SA No ID ID ]
sending packet: from 10.48.62.178[500] to 10.48.67.167[500] (196 bytes)
received packet: from 10.48.67.167[500] to 10.48.62.178[500] (180 bytes)
parsed QUICK_MODE response 1397274205 [ HASH SA No ID ID N((24576)) ]
connection 'ezvpn' established successfully
No leaks detected, 1 suppressed by whitelist

```

3. Quand vous activez met au point sur strongSwan, beaucoup d'informations peut être retourné. C'est le plus important mettent au point pour les utiliser quand le tunnel est initié :

```

#IKE Phase
06[CFG] received stroke: initiate 'ezvpn'
04[IKE] initiating Aggressive Mode IKE_SA ezvpn[1] to 10.48.67.167
03[CFG] proposal matches
03[CFG] received proposals: IKE:AES_CBC_128/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_1024
03[CFG] selected proposal: IKE:AES_CBC_128/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_1024
16[IKE] IKE_SA ezvpn[1] state change: CONNECTING => ESTABLISHED
16[IKE] scheduling reauthentication in 86210s

#Xauth phase
15[KNL] 10.48.62.178 is on interface eth1
15[IKE] installing new virtual IP 10.10.0.1
15[KNL] virtual IP 10.10.0.1 installed on eth1

#Ipsec
05[CFG] proposal matches
05[CFG] received proposals: ESP:AES_CBC_128/HMAC_SHA1_96/NO_EXT_SEQ
05[CFG] selected proposal: ESP:AES_CBC_128/HMAC_SHA1_96/NO_EXT_SEQ
05[KNL] adding SAD entry with SPI 7600acd8 and reqid

15[CFG] proposing traffic selectors for us:
15[CFG] 10.10.0.1/32
15[CFG] proposing traffic selectors for other:
15[CFG] 192.168.1.0/24

#Local settings
charon: 05[KNL] getting a local address in traffic selector 10.10.0.1/32
charon: 05[KNL] using host 10.10.0.1
charon: 05[KNL] using 10.48.62.129 as nexthop to reach 10.48.67.167
charon: 05[KNL] 10.48.62.178 is on interface eth1
charon: 05[KNL] installing route: 192.168.1.0/24 via 10.48.62.129 src 10.10.0.1
dev eth1
charon: 05[KNL] getting iface index for eth1
charon: 05[KNL] policy 10.10.0.1/32 === 192.168.1.0/24 out (mark 0/0x00000000)
already exists, increasing refcount
charon: 05[KNL] updating policy 10.10.0.1/32 === 192.168.1.0/24 out

```

#### 4. Envoyez le trafic du client :

```
gentool ~ # ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_req=1 ttl=255 time=1.19 ms
64 bytes from 192.168.1.1: icmp_req=2 ttl=255 time=1.19 ms
64 bytes from 192.168.1.1: icmp_req=3 ttl=255 time=1.12 ms
64 bytes from 192.168.1.1: icmp_req=4 ttl=255 time=1.16 ms
64 bytes from 192.168.1.1: icmp_req=4 ttl=255 time=1.26 ms
^C
--- 192.168.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.128/1.171/1.199/0.036 ms
```

#### 5. Vérifiez l'interface dynamique sur le logiciel de Cisco IOS :

```
Bsns-7200-2#sh int Virtual-Access1
Virtual-Access1 is up, line protocol is up
Hardware is Virtual Access interface
  Interface is unnumbered. Using address of GigabitEthernet0/1 (10.48.67.167)
  MTU 17878 bytes, BW 100000 Kbit/sec, DLY 50000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL
  Tunnel vaccess, cloned from Virtual-Templat1
  Vaccess status 0x4, loopback not set
  Keepalive not set
  Tunnel source 10.48.67.167 (GigabitEthernet0/1), destination 10.48.62.178
  Tunnel Subblocks:
    src-track:
      Virtual-Access1 source tracking subblock associated with
      GigabitEthernet0/1
      Set of tunnels with source GigabitEthernet0/1, 2 members (includes
      iterators), on interface <OK>
  Tunnel protocol/transport IPSEC/IP
  Tunnel TTL 255
  Tunnel transport MTU 1438 bytes
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "ipsecprof")
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 00:07:19
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    5 packets input, 420 bytes, 0 no buffer
    Received 0 broadcasts (0 IP multicasts)
    0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    5 packets output, 420 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 unknown protocol drops
    0 output buffer failures, 0 output buffers swapped out
```

#### 6. Vérifiez les compteurs d'IPSec sur le logiciel de Cisco IOS :

```
Bsns-7200-2#show crypto session detail
Crypto session current status
```



Code: C - IKE Configuration mode, D - Dead Peer Detection  
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation  
X - IKE Extended Authentication, F - IKE Fragmentation

Interface: Virtual-Access1

**Username: cisco**

**Profile: test**

**Group: RA**

**Assigned address: 10.10.0.1**

Uptime: 00:39:25

Session status: UP-ACTIVE

Peer: 10.48.62.178 port 500 fvrf: (none) ivrf: (none)

Phase1\_id: RA

Desc: (none)

IKEv1 SA: local 10.48.67.167/500 remote 10.48.62.178/500 Active

Capabilities:CDX connid:13002 lifetime:00:20:34

IPSEC FLOW: permit ip **192.168.1.0/255.255.255.0** host **10.10.0.1**

Active SAs: 2, origin: crypto map

Inbound: **#pkts dec'ed 5** drop 0 life (KB/Sec) KB Vol Rekey Disabled/1234

Outbound: **#pkts enc'ed 5** drop 0 life (KB/Sec) KB Vol Rekey Disabled/1234

## 7. Vérifiez l'état sur strongSwan :

```
gentool ~ # ipsec statusall
```

```
Status of IKE charon daemon (strongSwan 5.0.4, Linux 3.2.12-gentoo, x86_64):
```

```
  uptime: 41 minutes, since Jun 09 10:45:59 2013
```

```
  malloc: sbrk 1069056, mmap 0, used 896944, free 172112
```

```
  worker threads: 7 of 16 idle, 8/1/0/0 working, job queue: 0/0/0/0, scheduled: 2
```

```
  loaded plugins: charon aes des sha1 sha2 md5 random nonce x509 revocation
```

```
  constraints pubkey pkcs1 pkcs8 pgp dnskey pem openssl gcrypt fips-prf gmp
```

```
  xcbc cmac hmac attr kernel-netlink resolve socket-default stroke updown
```

```
  eap-identity eap-sim eap-aka eap-aka-3gpp2 eap-simaka-pseudonym
```

```
  eap-simaka-reauth eap-md5 eap-gtc eap-mschapv2 eap-radius xauth-generic dhcp
```

```
Listening IP addresses:
```

```
  192.168.0.10
```

```
  10.48.62.178
```

```
  2001:420:44ff:ff61:250:56ff:fe99:7661
```

```
  192.168.2.1
```

```
Connections:
```

```
  ezvpn: 10.48.62.178...10.48.67.167 IKEv1 Aggressive
```

```
  ezvpn: local: [RA] uses pre-shared key authentication
```

```
  ezvpn: local: [RA] uses XAuth authentication: any with XAuth identity
```

```
'cisco'
```

```
  ezvpn: remote: [10.48.67.167] uses pre-shared key authentication
```

```
  ezvpn: child: dynamic === 192.168.1.0/24 TUNNEL
```

```
Security Associations (1 up, 0 connecting):
```

```
  ezvpn[1]: ESTABLISHED 41 minutes ago, 10.48.62.178[RA]...
```

```
10.48.67.167[10.48.67.167]
```

```
  ezvpn[1]: IKEv1 SPIs: 0fa722d2f09bffe0_i* 6b4c44bae512b278_r, pre-shared  
key+XAuth reauthentication in 23 hours
```

```
  ezvpn[1]: IKE proposal: AES_CBC_128/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_1024
```

```
  ezvpn{1}: INSTALLED, TUNNEL, ESP SPIs: c805b9ba_i 7600acd8_o
```

```
  ezvpn{1}: AES_CBC_128/HMAC_SHA1_96, 420 bytes_i (5 pkts, 137s ago), 420  
bytes_o (5 pkts, 137s ago), rekeying in 13 minutes
```

```
  ezvpn{1}: 10.10.0.1/32 === 192.168.1.0/24
```

```
No leaks detected, 1 suppressed by whitelist
```

## Dépannez

Il n'existe actuellement aucune information de dépannage spécifique pour cette configuration.

# Résumé

Ce document a décrit la configuration d'un client strongSwan qui se connecte en tant que client vpn d'IPSec au logiciel de Cisco IOS.

Il est également possible de configurer un tunnel entre réseaux locaux d'IPSec entre le logiciel de Cisco IOS et strongSwan. Supplémentaire, IKEv2 entre les deux périphériques fonctionne correctement chacun des deux pour le distant et l'accès d'entre réseaux locaux.

## Informations connexes

- [Documentation d'Openswan](#)
- [Documentation utilisateur de StrongSwan](#)
- [Configuration de la version 2 d'échange de clés Internet \(IKE\) et de la section de site à site de FlexVPN de FlexVPN et de guide de configuration de version 2 d'échange de clés Internet \(IKE\), version de Cisco IOS 15M&T](#)
- [Support et documentation techniques - Cisco Systems](#)