IPSec - Connexion entre un pare-feu PIX et un client VPN Cisco avec caractères génériques, clés pré-partagées, mode configuration avec authentification étendue

Contenu

Introduction

Cet exemple de configuration montre comment connecter un client VPN à un pare-feu PIX à l'aide de caractères génériques, mode-config, la commande **sysopt connection permit-ipsec** et l'authentification étendue (Xauth).

Afin de voir la configuration TACACS+ et RADIUS pour PIX 6.3 et versions ultérieures, référez-vous à TACACS+ et RADIUS pour PIX 6.3 et PIX/ASA 7.x Exemple de configuration.

Le client VPN prend en charge la norme AES (Advanced Encryption Standard) en tant qu'algorithme de chiffrement dans Cisco VPN Client version 3.6.1 et ultérieure et avec PIX Firewall 6.3. Le client VPN prend en charge des tailles de clés de 128 bits et 256 bits uniquement. Pour plus d'informations sur la configuration d'AES, référez-vous à Comment configurer le client VPN Cisco sur PIX avec AES.

Référez-vous à Exemple de configuration d'authentification RADIUS PIX/ASA 7.x et Cisco VPN Client 4.x pour Windows avec Microsoft Windows 2003 IAS pour configurer la connexion VPN d'accès à distance entre un client VPN Cisco (4.x pour Windows) et le dispositif de sécurité de la gamme PIX 500 7.x à l'aide d'un serveur RADIUS Microsoft Windows 200000000000000000000000000000000000000000000000000000000000000000000000

Consultez Exemple de configuration d'IPsec entre un concentrateur VPN 3000 et un client VPN 4.x pour Windows à l'aide de RADIUS pour l'authentification et la comptabilisation des utilisateurs pour établir un tunnel IPsec entre un concentrateur Cisco VPN 3000 et un Client VPN Cisco 4.x pour Windows utilisant RADIUS pour la vérification de l'ID de l'utilisateur et la gestion des comptes.

Consultez Configuration d'IPSec entre un routeur Cisco IOS et un client VPN Cisco 4.x pour Windows à l'aide de RADIUS pour l'authentification d'utilisateur pour configurer une connexion entre un routeur et le Client VPN Cisco 4.x utilisant RADIUS pour la vérification de l'ID de l'utilisateur.

Conditions préalables

Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

Components Used

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

- Client VPN Cisco 4.x Ce produit possède des fonctionnalités VPN avancées, contrairement au client VPN sécurisé Cisco 1.x.
- PIX Firewall 515E version 6.3(3).

Remarque : La technologie de chiffrement est soumise à des contrôles d'exportation. Il est de votre responsabilité de connaître la loi relative à l'exportation des technologies de chiffrement. Pour de plus amples renseignements, consultez le site Web du Bureau de l'administration des exportations. Si vous avez des questions concernant le contrôle des exportations, veuillez envoyer un courriel à export@cisco.com.

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. If your network is live, make sure that you understand the potential impact of any command.

Conventions

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

Informations générales

La commande **sysopt connection permit-ipsec** autorise implicitement tout paquet provenant d'un tunnel IPsec à contourner la vérification d'une **liste d'accès** associée, d'un **conduit** ou d'une commande **access-group** pour les connexions IPsec. Xauth authentifie l'utilisateur IPsec sur un serveur TACACS+ ou RADIUS externe. En plus de la clé pré-partagée générique, l'utilisateur doit fournir un nom d'utilisateur/mot de passe.

Un utilisateur ayant un client VPN reçoit une adresse IP de son FAI. Cette adresse est remplacée par une adresse IP du pool d'adresses IP du PIX. L'utilisateur a accès à tout ce qui se trouve à l'intérieur du pare-feu, y compris les réseaux. Les utilisateurs qui n'exécutent pas le client VPN peuvent se connecter uniquement au serveur Web à l'aide de l'adresse externe fournie par l'affectation statique.

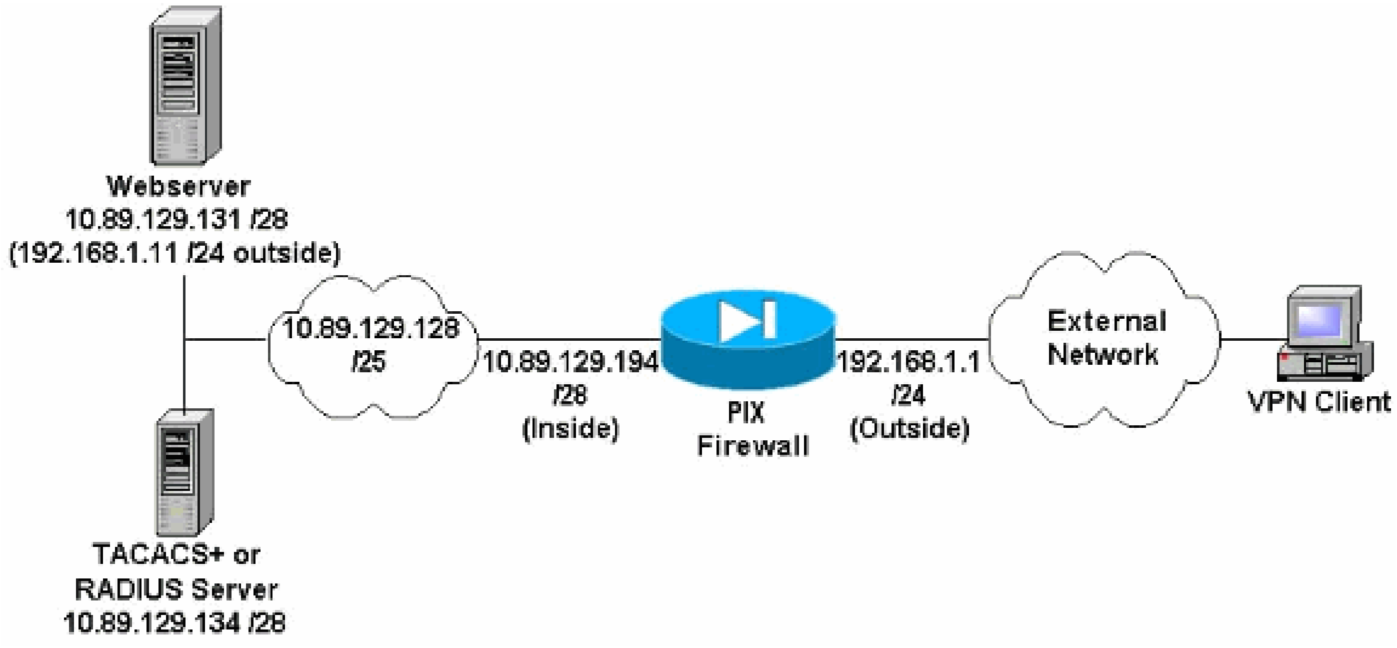
Configuration

Cette section vous fournit des informations pour configurer les fonctionnalités décrites dans ce document.

Remarque : Utilisez l'outil de recherche de commandes (clients inscrits seulement) pour en savoir plus sur les commandes figurant dans le présent document.

Diagramme du réseau

Ce document utilise la configuration réseau suivante :



Notes de diagramme de réseau

- Les hôtes Internet qui accèdent au serveur Web à l'aide de l'adresse IP globale 192.168.1.1 sont authentifiés même si une connexion VPN n'est pas établie. Ce trafic *n*'est *pas* chiffré.
- Les clients VPN peuvent accéder à tous les hôtes du réseau interne (10.89.129.128 /25) une fois leur tunnel IPsec établi. Tout le trafic du client VPN au pare-feu PIX est chiffré. Sans tunnel IPsec, ils ne peuvent accéder au serveur Web que par l'intermédiaire de son adresse IP globale, mais doivent toujours s'authentifier.
- Les clients VPN proviennent d'Internet et leurs adresses IP ne sont pas connues à l'avance.

Configurations

Ce document utilise les configurations suivantes.

- Configuration PIX 6.3(3)
- Configuration du client VPN 4.0.5
- Configuration VPN Client 3.5
- Configuration VPN Client 1.1

Configuration PIX 6.3(3)

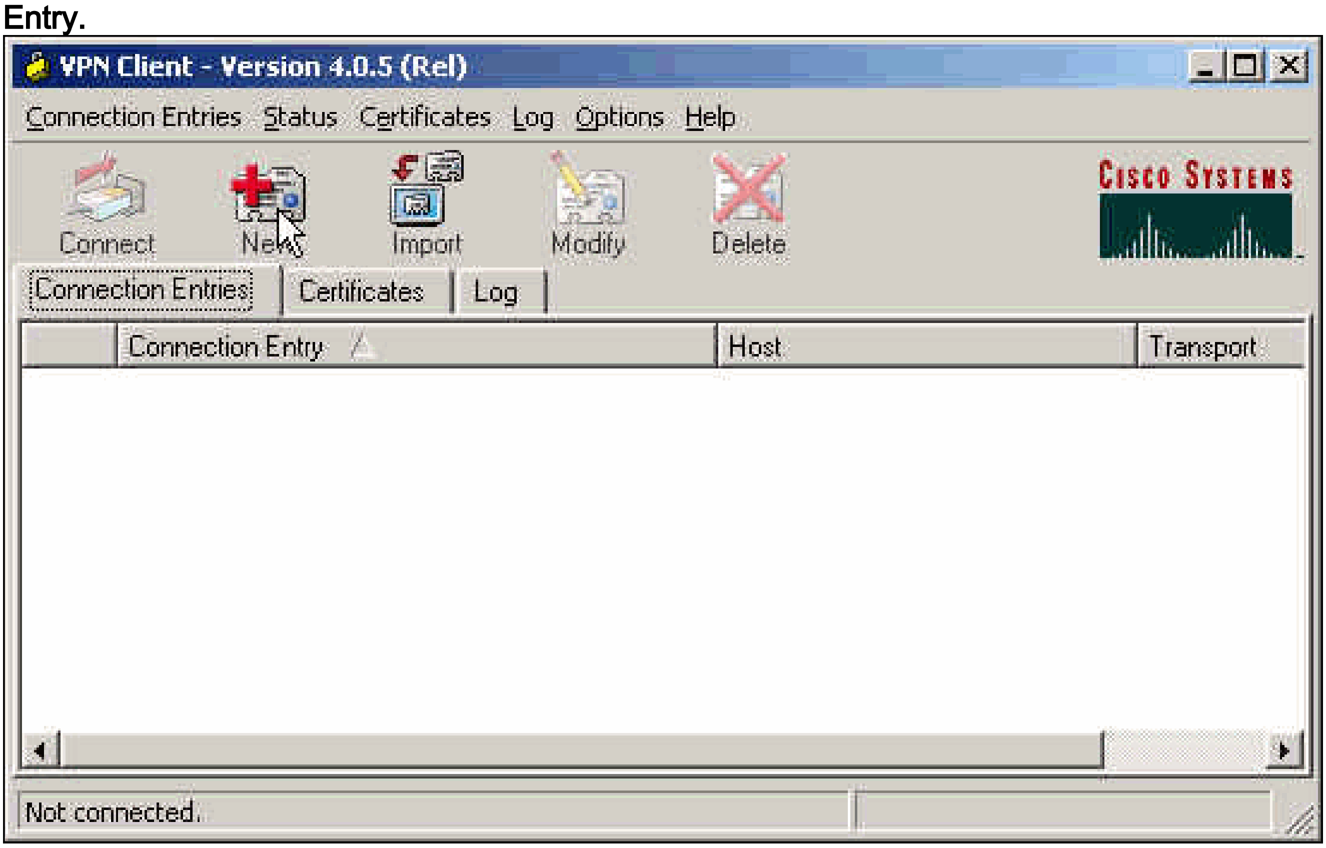
```
pixfirewall#show run
: Saved
:
PIX Version 6.3(3)
interface ethernet0 100full
interface ethernet1 100full
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pixfirewall
fixup protocol dns maximum-length 512
fixup protocol ftp 21
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol http 80
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol sip 5060
fixup protocol sip udp 5060
fixup protocol skinny 2000
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol tftp 69
names
!--- Do not use Network Address Translation (NAT) for
inside-to-pool !--- traffic. This should not go through
NAT. access-list 101 permit ip 10.89.129.128
255.255.255.240 10.89.129.192 255.255.255.240 !---
Permits Internet Control Message Protocol (ICMP) !---
Transmission Control Protocol (TCP) and User Datagram
Protocol (UDP) !--- traffic from any host on the
Internet (non-VPN) to the web server. access-list 120
permit icmp any host 10.89.129.131 access-list 120
permit tcp any host 10.89.129.131 access-list 120 permit
udp any host 10.89.129.131 pager lines 24 mtu outside
1500 mtu inside 1500 ip address outside 192.168.1.1
255.255.255.0 ip address inside 10.89.129.194
255.255.255.240 ip audit info action alarm ip audit
attack action alarm !--- Specifies the inside IP address
range to be assigned !--- to the VPN Clients. ip local
pool VPNpool 10.89.129.200-10.89.129.204 no failover
failover timeout 0:00:00 failover poll 15 no failover ip
address outside no failover ip address inside pdm
history enable arp timeout 14400 !--- Defines a pool of
global addresses to be used by NAT. global (outside) 1
192.168.1.6-192.168.1.10 nat (inside) 0 access-list 101
nat (inside) 1 0.0.0.0 0.0.0.0 0 0 !--- Specifies which
outside IP address to apply to the web server. static
(inside,outside) 192.168.1.11 10.89.129.131 netmask
255.255.255.255 0 0 !--- Apply ACL 120 to the outside
interface in the inbound direction. access-group 120 in
interface outside !--- Defines a default route for the
PIX. route outside 0.0.0.0 0.0.0.0 192.168.1.3 1 !---
Defines a route for traffic within the PIX's !--- subnet
to reach other inside hosts. route inside 10.89.129.128
255.255.255.128 10.89.129.193 1 timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc
0:10:00 h225 1:00:00 timeout h323 0:05:00 mgcp 0:05:00
sip 0:30:00 sip_media 0:02:00 timeout uauth 0:05:00
absolute aaa-server TACACS+ protocol tacacs+ aaa-server
RADIUS protocol radius aaa-server LOCAL protocol local
!--- Authentication, authorization, and accounting (AAA)
statements !--- for authentication. !--- Use either of
these statements to define the protocol of the group
AuthInbound. !--- You cannot use both.
aaa-server AuthInbound protocol tacacs+

!--- OR aaa-server AuthInbound protocol radius !---
After you define the protocol of the group AuthInbound,
define !--- a server of the same type. !--- In this case
we specify the TACACS+ server and key of "secretkey".
aaa-server AuthInbound (inside) host 10.89.129.134
secretkey timeout 10 !--- Authenticate HTTP, FTP, and
Telnet traffic to the web server. aaa authentication
include http outside 10.89.129.131 255.255.255.255
0.0.0.0 0.0.0.0 AuthInbound aaa authentication include
ftp outside 10.89.129.131 255.255.255.255 0.0.0.0
0.0.0.0 AuthInbound aaa authentication include telnet
outside 10.89.129.131 255.255.255.255 0.0.0.0 0.0.0.0
AuthInbound no snmp-server location no snmp-server
contact snmp-server community public no snmp-server
enable traps floodguard enable !--- Trust IPsec traffic
and avoid going through ACLs/NAT. sysopt connection
permit-ipsec !--- IPsec and dynamic map configuration.
crypto ipsec transform-set myset esp-des esp-md5-hmac
crypto dynamic-map dynmap 10 set transform-set myset
crypto map mymap 10 ipsec-isakmp dynamic dynmap !---
Assign IP address for VPN 1.1 Clients. crypto map mymap
client configuration address initiate crypto map mymap
client configuration address respond !--- Use the AAA
server for authentication (AuthInbound). crypto map
mymap client authentication AuthInbound !--- Apply the
IPsec/AAA/ISAKMP configuration to the outside interface.
crypto map mymap interface outside isakmp enable outside
!--- Pre-shared key for VPN 1.1 Clients. isakmp key
******** address 0.0.0.0 netmask 0.0.0.0 isakmp identity
address !--- Assign address from "VPNpool" pool for VPN
1.1 Clients. isakmp client configuration address-pool
local VPNpool outside !--- ISAKMP configuration for VPN
Client 3.x/4.x. isakmp policy 10 authentication pre-
share isakmp policy 10 encryption des isakmp policy 10
hash md5 isakmp policy 10 group 2 isakmp policy 10
lifetime 86400 !--- ISAKMP configuration for VPN Client
1.x. isakmp policy 20 authentication pre-share isakmp
policy 20 encryption des isakmp policy 20 hash md5
isakmp policy 20 group 1 isakmp policy 20 lifetime 86400
!--- Assign addresses from "VPNpool" for VPN Client
3.x/4.x. vpngroup vpn3000 address-pool VPNpool vpngroup
vpn3000 idle-time 1800 !--- Group password for VPN
Client 3.x/4.x (not shown in configuration). vpngroup
vpn3000 password ******** telnet timeout 5 ssh timeout 5
console timeout 0 terminal width 80
Cryptochecksum:ba54c063d94989cbd79076955dbfeefc : end
pixfirewall#
```

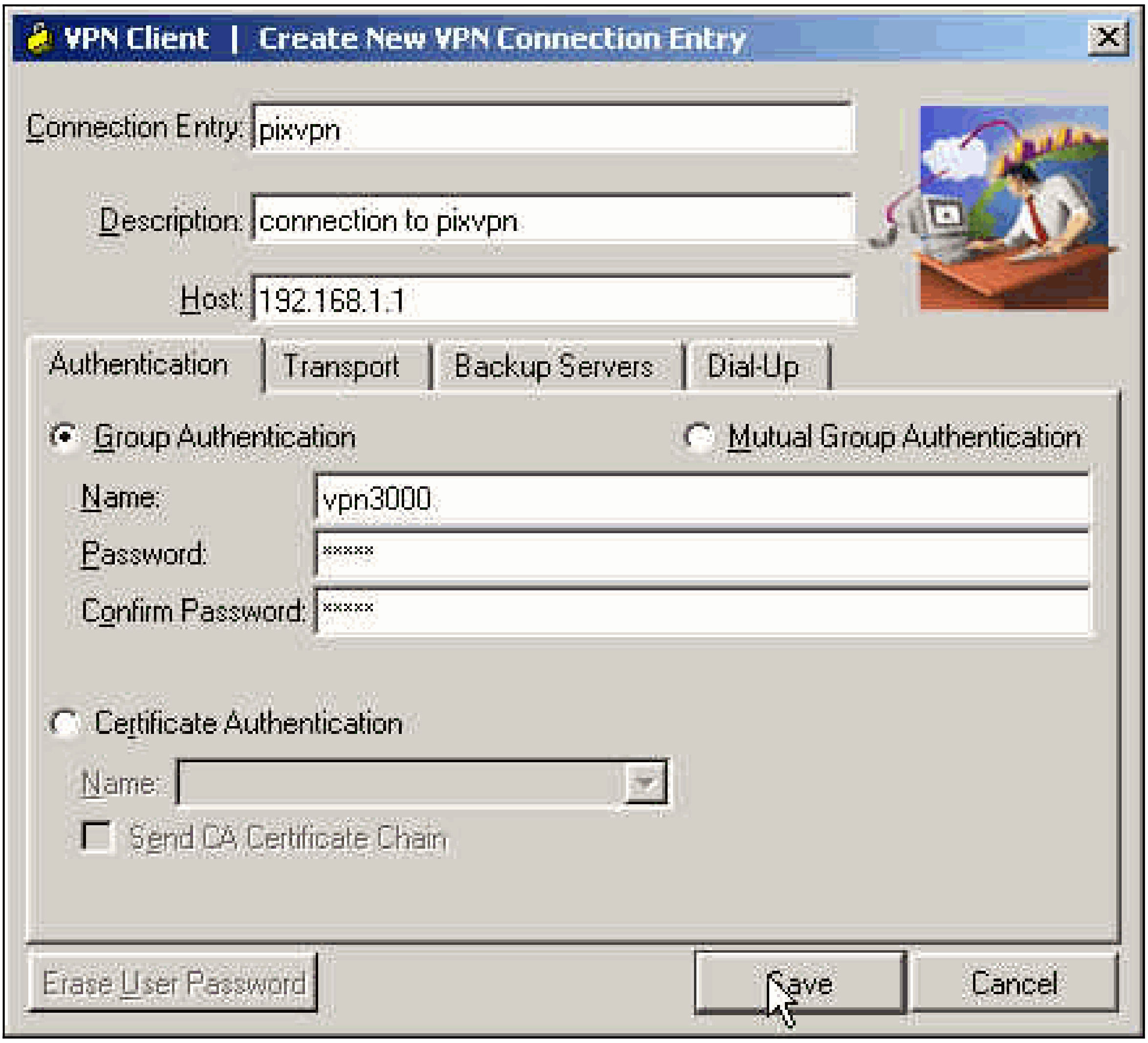
Configuration du client VPN 4.0.5

Exécutez ces étapes pour configurer le client VPN 4.0.5.

1. Sélectionnez **Démarrer > Programmes > Client VPN Cisco Systems > Client VPN**.
2. Cliquez **New pour ouvrir la fenêtre Create New VPN Connection Entry.**



3. Entrez le nom de l'entrée de connexion avec une description. Entrez l'adresse IP externe du pare-feu PIX dans la zone Host. Entrez ensuite le nom du groupe VPN et le mot de passe, puis cliquez sur



Connection Entry: pixvpn

Description: connection to pixvpn

Host: 192.168.1.1

● Group Authentication ○ Mutual Group Authentication

Name: vpn3000

Password: *****

Confirm Password: *****

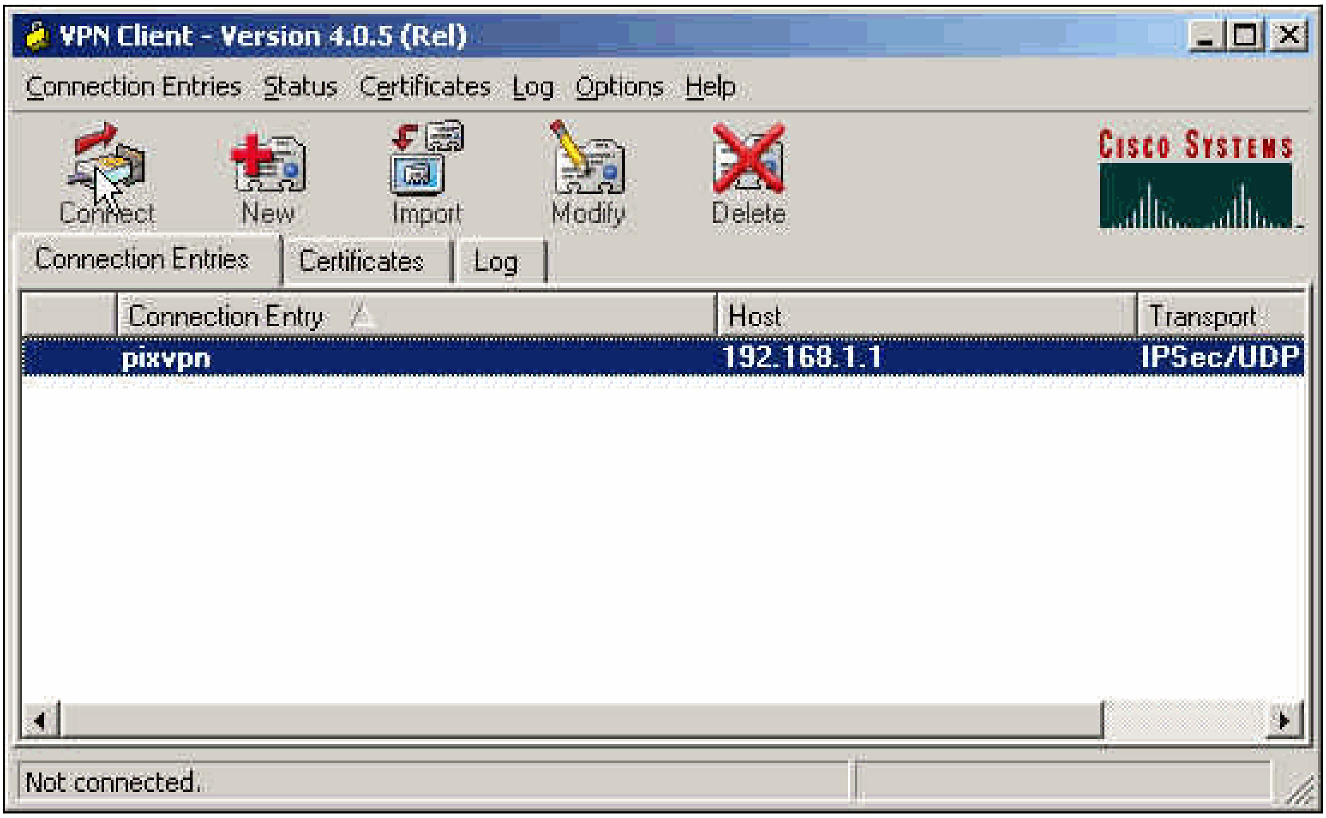
○ Certificate Authentication

Name:

□ Send CA Certificate Chain

Save.

4. Dans la fenêtre principale du client VPN, cliquez sur la connexion que vous souhaitez utiliser et cliquez sur le bouton **Connect**.

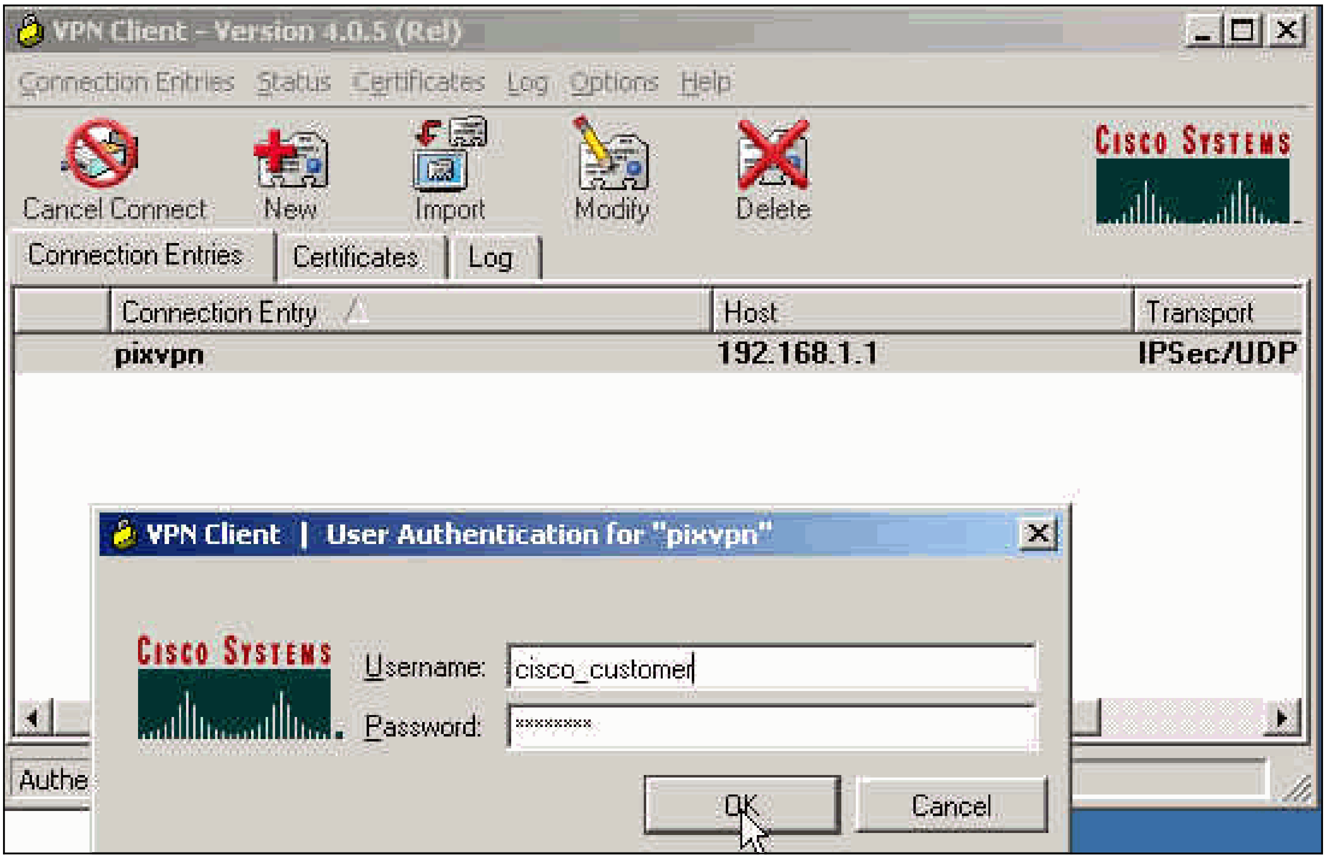


Connection Entry	Host	Transport
pixvpn	192.168.1.1	IPSec/UDP

Not connected.

5. Lorsque vous y êtes invité, saisissez le nom d'utilisateur et le mot de passe pour Xauth et

cliquez sur **OK pour vous connecter au réseau distant.**


VPN Client - Version 4.0.5 (Rel)
Connection Entries Status Certificates Log Options Help
Cancel Connect
New
Import
Modify
Delete
Cisco Systems
Connection Entries
Certificates
Log
Authe
VPN Client | User Authentication for "pixvpn"
OK
Cancel

Connection Entry	Host	Transport
pixvpn	192.168.1.1	IPSec/UDP

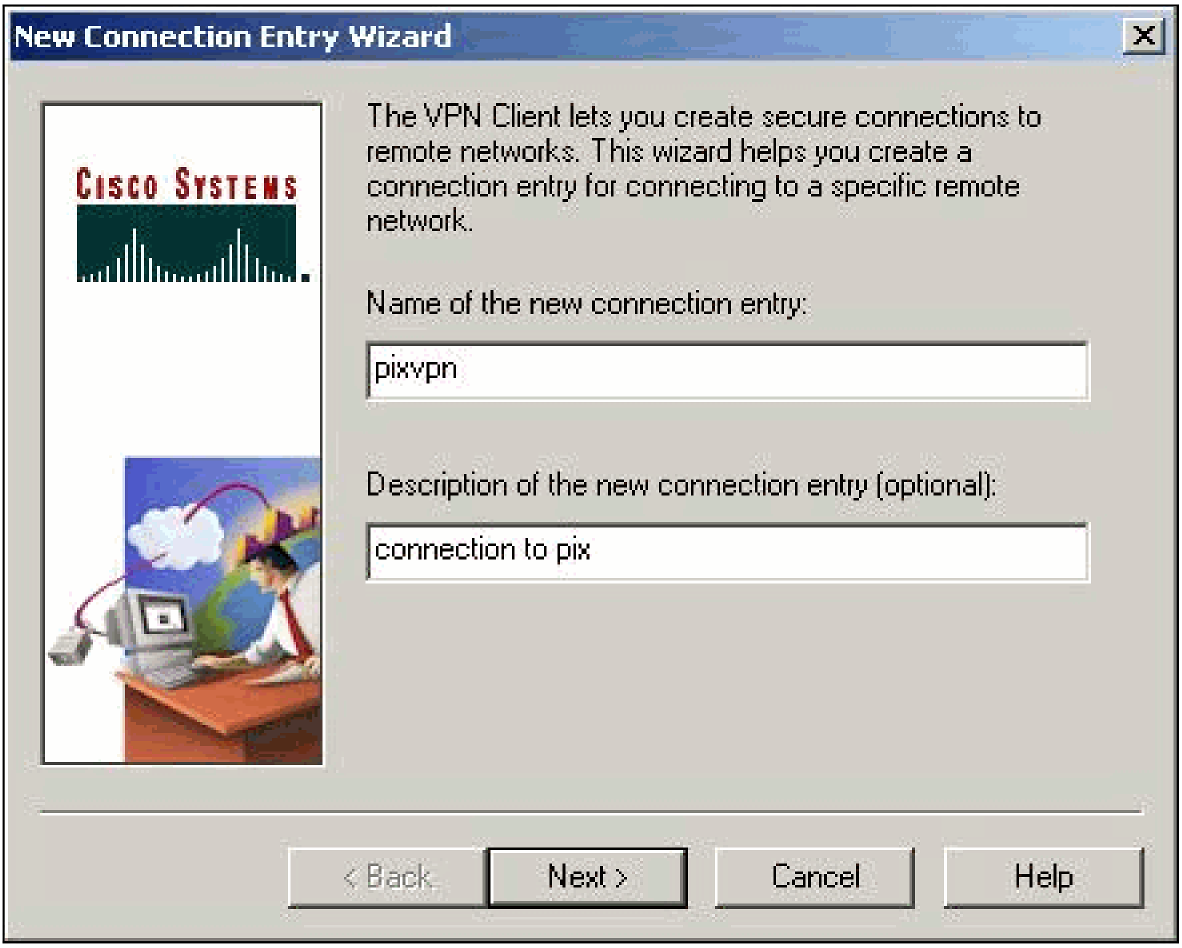
Username: cisco_customer

Password: ********

Configuration VPN Client 3.5

Complétez ces étapes pour configurer le client VPN 3.5.

1. Sélectionnez **Démarrer > Programmes > Cisco Systems VPN Client > VPN Dialer**.
2. Cliquez sur **Nouveau** pour lancer l'Assistant Nouvelle entrée de connexion.
3. Entrez le nom de votre nouvelle entrée de connexion et cliquez sur



New Connection Entry Wizard

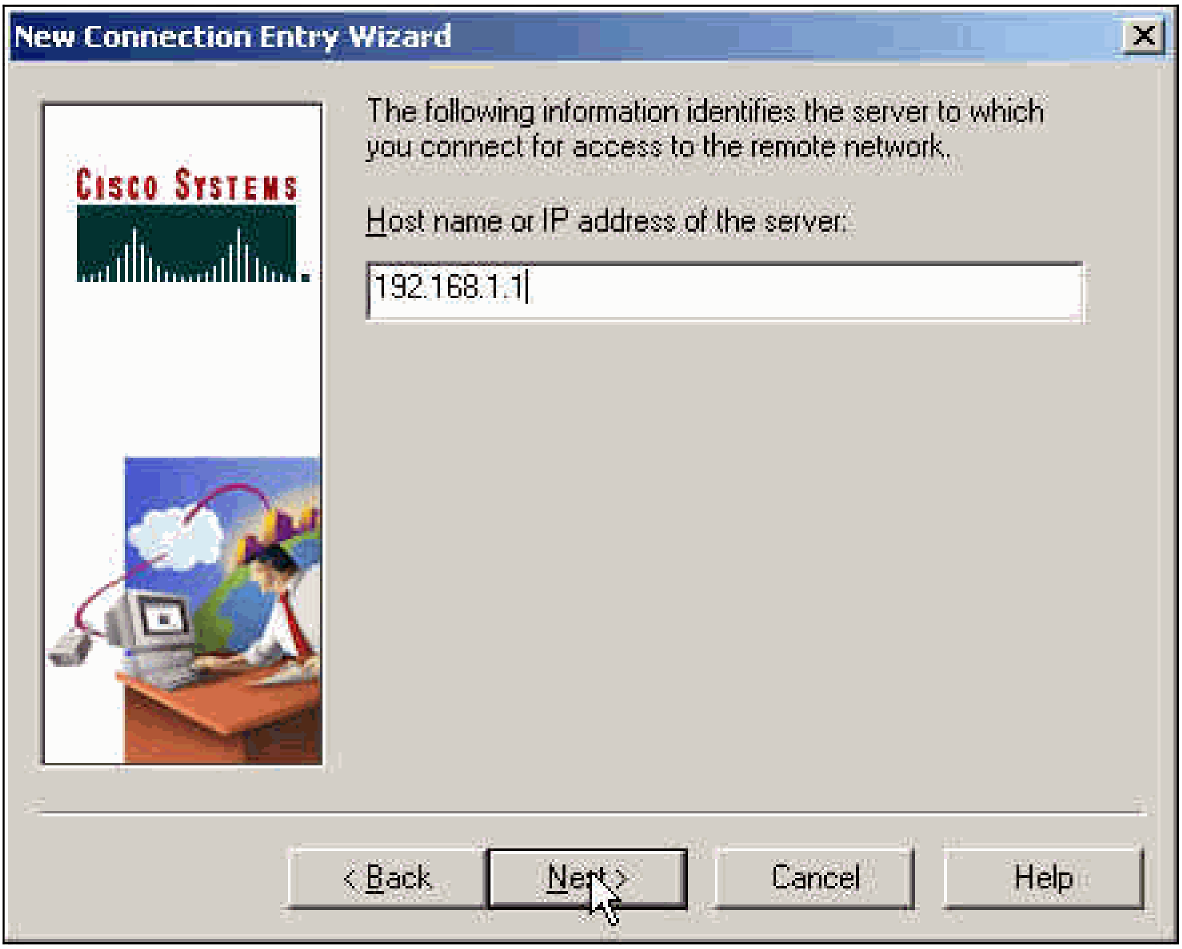
The VPN Client lets you create secure connections to remote networks. This wizard helps you create a connection entry for connecting to a specific remote network.

Name of the new connection entry: pixvpn

Description of the new connection entry (optional): connection to pix

Suivant.

4. Entrez le nom d'hôte ou l'adresse IP du serveur utilisé pour se connecter au serveur distant et cliquez sur



New Connection Entry Wizard

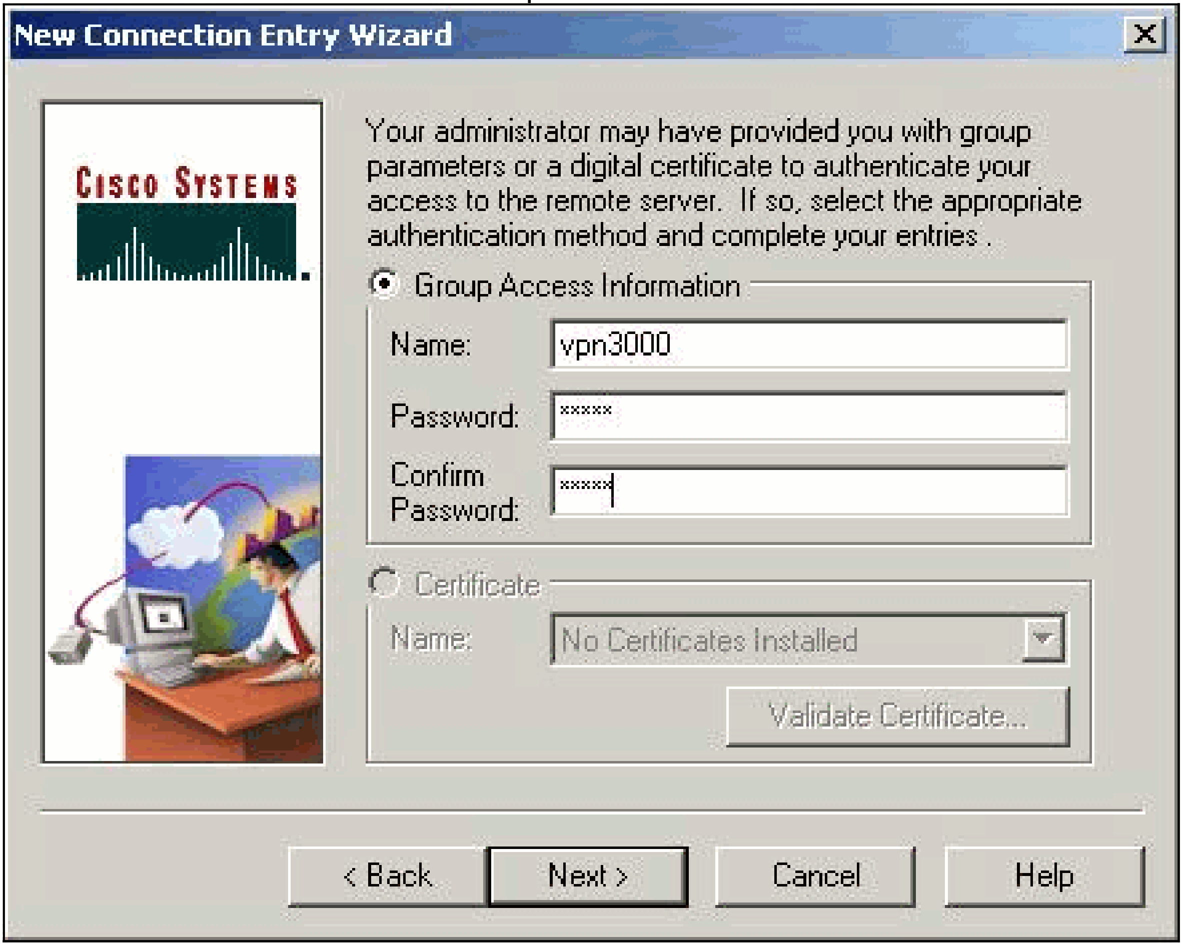
The following information identifies the server to which you connect for access to the remote network.

Host name or IP address of the server: 192.168.1.1

Suivant.

5. Sélectionnez **Group Access Information** et saisissez le nom et le mot de passe utilisés pour

authentifier votre accès au serveur distant. Cliquez sur **Next** (Suivant).



New Connection Entry Wizard

Your administrator may have provided you with group parameters or a digital certificate to authenticate your access to the remote server. If so, select the appropriate authentication method and complete your entries .

● Group Access Information

Name: vpn3000

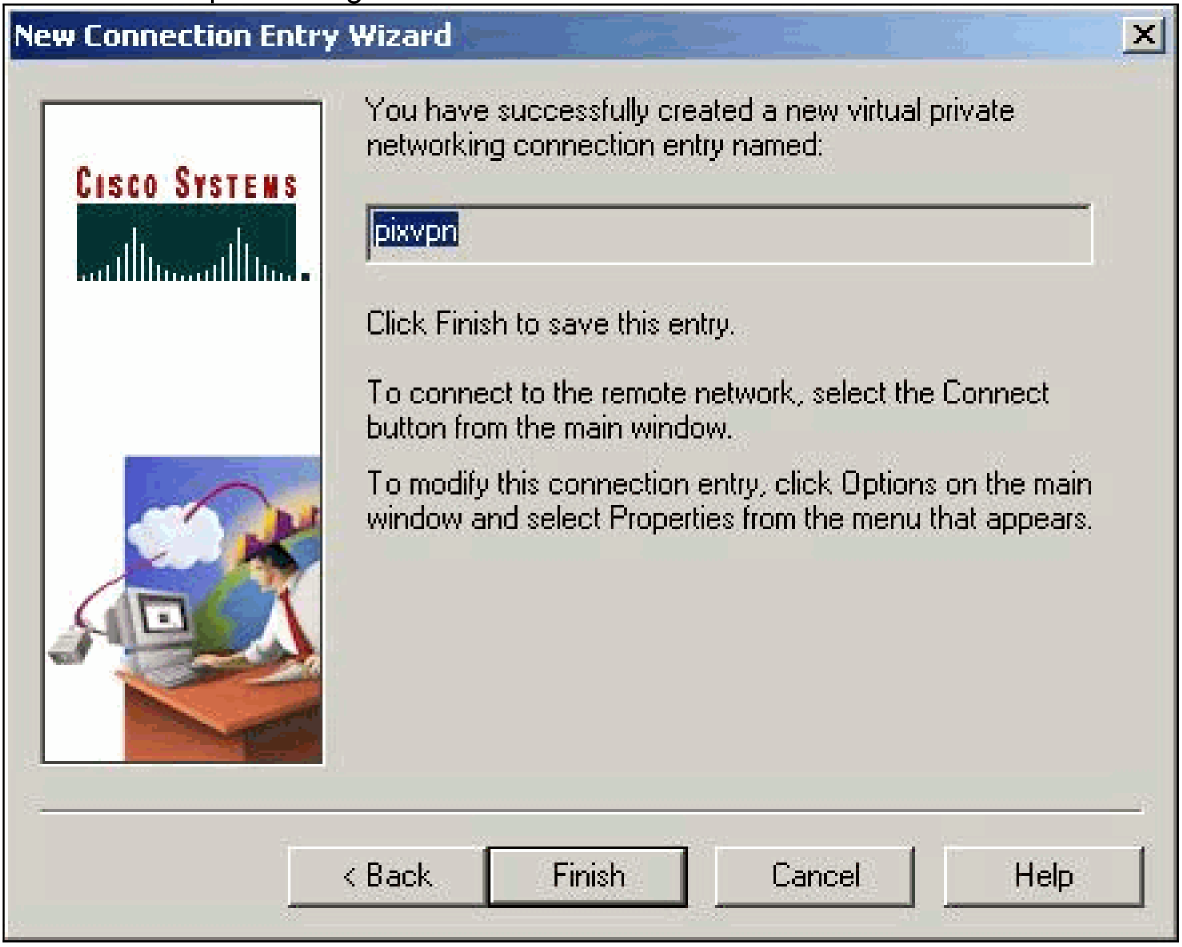
Password: *****

Confirm Password: *****

○ Certificate

Name: No Certificates Installed

6. Cliquez sur **Terminer** pour enregistrer la nouvelle entrée.



New Connection Entry Wizard

You have successfully created a new virtual private networking connection entry named:

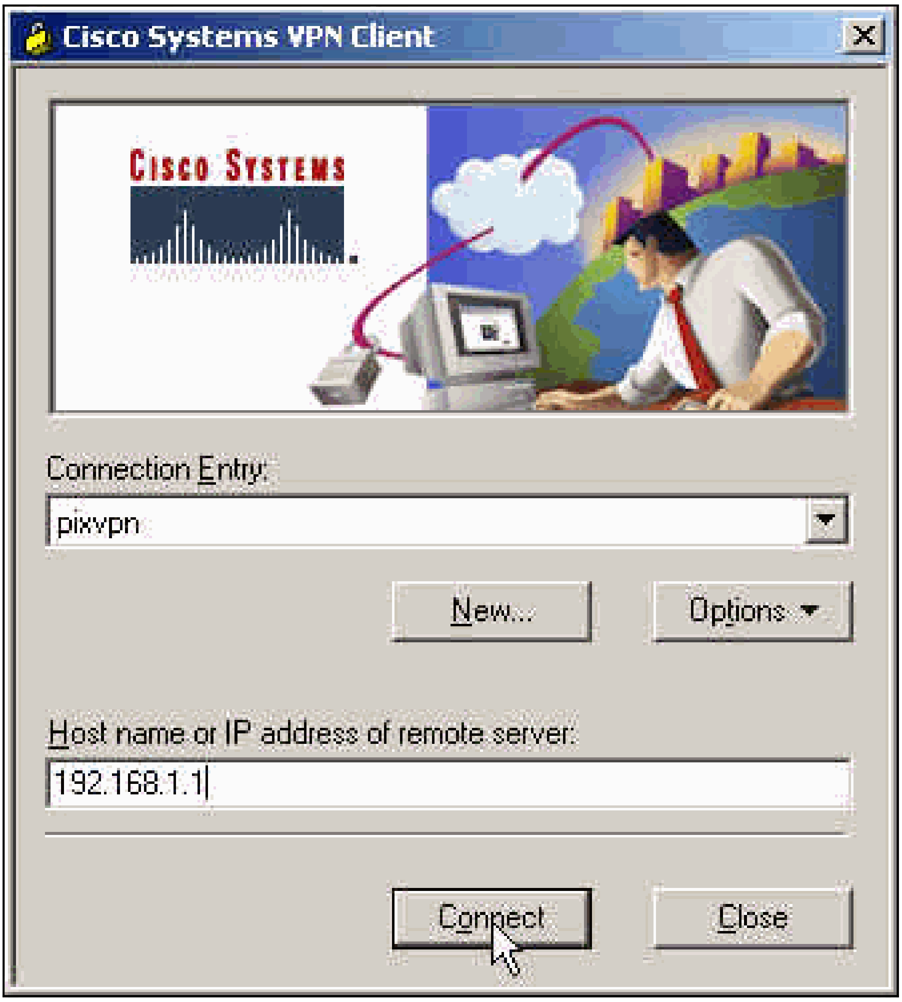
pixvpn

Click Finish to save this entry.

To connect to the remote network, select the Connect button from the main window.

To modify this connection entry, click Options on the main window and select Properties from the menu that appears.

7. Sélectionnez l'entrée de connexion dans le numéroteur et cliquez sur



Cisco Systems VPN Client

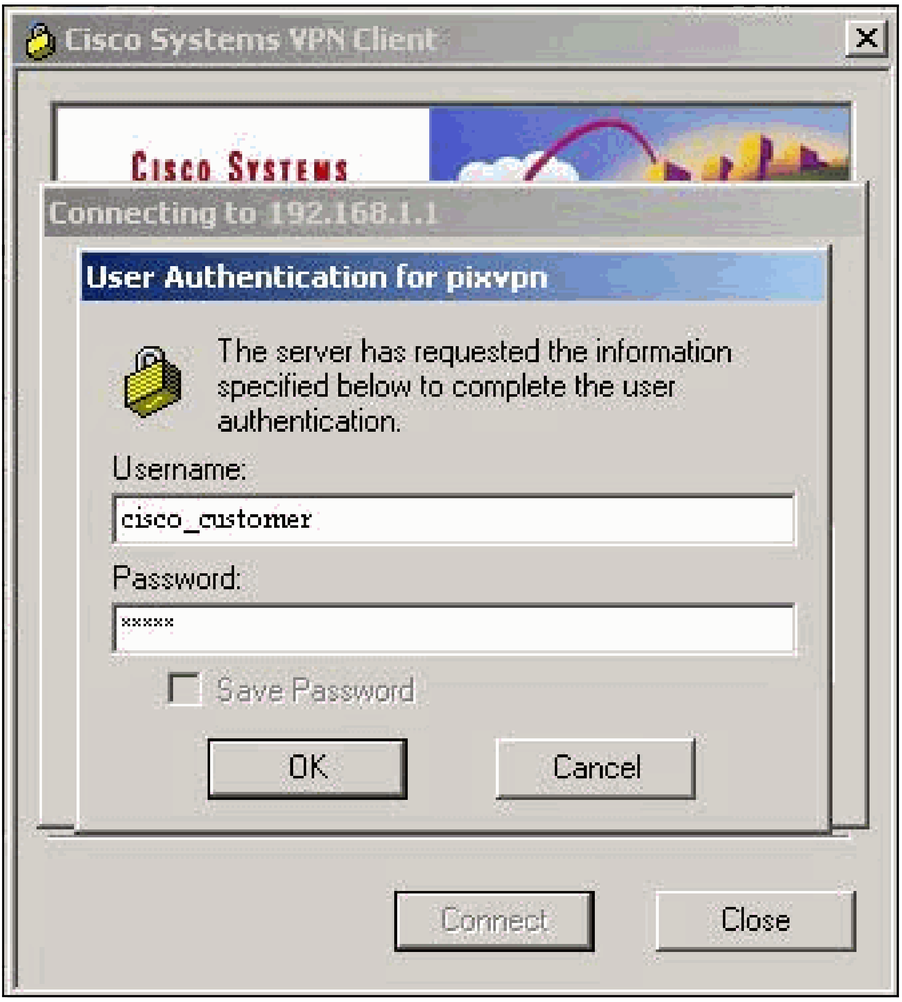
Connection Entry: pixvpn

Host name or IP address of remote server: 192.168.1.1

Connect.

8. Lorsque vous y êtes invité, saisissez le nom d'utilisateur et le mot de passe pour Xauth et cliquez sur **OK pour vous connecter au réseau**

distant.



User Authentication for pixvpn

The server has requested the information specified below to complete the user authentication.

Username: cisco_customer

Password: *****

□ Save Password

Configuration VPN Client 1.1

```
Network Security policy:
 1- TACconn
     My Identity
          Connection security: Secure
          Remote Party Identity and addressing
          ID Type: IP subnet
          10.89.129.128
          255.255.255.128
          Port all Protocol all


     Connect using secure tunnel

          ID Type: IP address
          192.168.1.1


     Pre-shared Key=cisco1234


     Authentication (Phase 1)

     Proposal 1
         Authentication method: pre-shared key
         Encryp Alg: DES
         Hash Alg: MD5
         SA life: Unspecified
         Key Group: DH 1
     Key exchange (Phase 2)

     Proposal 1
         Encapsulation ESP
         Encrypt Alg: DES
         Hash Alg: MD5
         Encap: tunnel
         SA life: Unspecified
         no AH

 2- Other Connections
        Connection security: Non-secure
        Local Network Interface
          Name: Any
          IP Addr: Any
          Port: All
```

Ajoutez la gestion des comptes

La syntaxe de la commande à ajouter à la comptabilité est la suivante :

```
aaa accounting include acctg_service inbound|outbound l_ip l_mask [f_ip f_mask] server_tag
```

Par exemple, dans la configuration PIX, cette commande est ajoutée :

```
aaa accounting include any inbound
0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

Remarque : La commande **sysopt connection permit-ipsec**, et non la commande **sysopt ipsec pl-compatible**, est nécessaire pour que la comptabilité Xauth fonctionne. La comptabilité Xauth ne fonctionne pas uniquement avec la commande **sysopt ipsec pl-compatible**. La comptabilité Xauth est valide pour les connexions TCP, pas ICMP ou UDP.

Ce résultat est un exemple d'enregistrements comptables TACACS+ :

```
07/27/2004 15:17:54 cisco_customer Default Group 10.89.129.200 stop 15 .. 99 1879 .. ..
   0x5 .. PIX 10.89.129.194 telnet
07/27/2004 15:17:39 cisco_customer Default Group 10.89.129.200 start .. .. .. .. .. ..
   0x5 .. PIX 10.89.129.194 telnet
```

Vérification

Référez-vous à cette section pour vous assurer du bon fonctionnement de votre configuration.

L'Outil Interpréteur de sortie (clients enregistrés uniquement) (OIT) prend en charge certaines commandes show. Utilisez l'OIT pour afficher une analyse de la sortie de la commande **show .**

Remarque : Consulter les renseignements importants sur les commandes de débogage avant d’utiliser les commandes de **débogage**.

Activez Cisco Secure Log Viewer afin de voir les débogages côté client.

- **debug crypto ipsec** - Utilisé pour voir les négociations IPsec de la phase 2.
- **debug crypto isakmp** - Utilisé pour voir les négociations ISAKMP de la phase 1.

Dépannage

Cette section fournit des informations que vous pouvez utiliser pour dépanner votre configuration. L'exemple de sortie Debug est également affiché.

Dépannage des commandes

L'Outil Interpréteur de sortie (clients enregistrés uniquement) (OIT) prend en charge certaines commandes show. Utilisez l'OIT pour afficher une analyse de la sortie de la commande **show .**

Remarque : Consulter les renseignements importants sur les commandes de débogage avant d'utiliser les commandes de **débogage**.

- **debug crypto engine** - Utilisé pour déboguer le processus crypto engine.

Exemple de débogage PIX

```
pixfirewall#show debug
debug crypto ipsec 1
debug crypto isakmp 1
debug crypto engine
debug fover status
     tx      Off
     rx      Off
     open    Off
     cable   Off
     txdmp   Off
     rxdmp   Off
     ifc     Off
     rxip    Off
     txip    Off
     get     Off
     put     Off
     verify  Off
     switch  Off
     fail    Off
     fmsg    Off
```

Débogues avec VPN Client 4.x

```
pixfirewall#
crypto_isakmp_process_block: src 192.168.1.2, dest 192.168.1.1
VPN Peer: ISAKMP: Added new peer: ip:192.168.1.2
Total VPN Peers:1
VPN Peer: ISAKMP: Peer ip:192.168.1.2 Ref cnt incremented
to:1 Total VPN Peers:1
OAK_AG exchange
ISAKMP (0): processing SA payload. message ID = 0

ISAKMP (0): Checking ISAKMP transform 1 against priority 10 policy
ISAKMP:     encryption 3DES-CBC
```

```
ISAKMP:      hash SHA
ISAKMP:      default group 2
ISAKMP:      extended auth pre-share
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of 0x0 0x20 0xc4 0x9b
ISAKMP (0): atts are not acceptable. Next payload is 3
ISAKMP (0): Checking ISAKMP transform 2 against priority 10 policy
ISAKMP:      encryption 3DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 2
ISAKMP:      extended auth pre-share
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of 0x0 0x20 0xc4 0x9b
ISAKMP (0): atts are not acceptable. Next payload is 3
ISAKMP (0): Checking ISAKMP transform 3 against priority 10 policy
ISAKMP:      encryption 3DES-CBC
ISAKMP:      hash SHA
ISAKMP:      default group 2
ISAKMP:      auth pre-shared
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of 0x0 0x20 0xc4 0x9b
ISAKMP (0): atts are not acceptable. Next payload is 3
ISAKMP (0): Checking ISAKMP transform 4 against priority 10 policy
ISAKMP:      encryption 3DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 2
ISAKMP:      auth pre-share
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of 0x0 0x20 0xc4 0x9b
ISAKMP (0): atts are not acceptable. Next payload is 3
ISAKMP (0): Checking ISAKMP transform 5 against priority 10 policy
ISAKMP:      encryption DES-CBC
ISAKMP:      hash SHA
ISAKMP:      default group 2
ISAKMP:      extended auth pre-share
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of 0x0 0x20 0xc4 0x9b
ISAKMP (0): atts are not acceptable. Next payload is 3
ISAKMP (0): Checking ISAKMP transform 6 against priority 10 policy
ISAKMP:      encryption DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 2
ISAKMP:      extended auth pre-share
ISAKMP:      life type in seconds
ISAKMP:      life duration (VPI) of 0x0 0x20 0xc4 0x9b
ISAKMP (0): atts are acceptable. Next payload is 3
!--- Attributes offered by the VPN Client are accepted by the PIX. ISAKMP (0): processing KE
payload. message ID = 0 ISAKMP (0): processing NONCE payload. message ID = 0 ISAKMP (0):
processing ID payload. message ID = 0 ISAKMP (0): processing vendor id payload ISAKMP (0):
processing vendor id payload ISAKMP (0): remote peer supports dead peer detection ISAKMP (0):
processing vendor id payload ISAKMP (0): speaking to a Unity client ISAKMP (0): ID payload next-
payload: 10 type : 1 protocol : 17 port : 500 length : 8 ISAKMP (0) : Total payload length: 12
return status is IKMP_NO_ERROR crypto_isakmp_process_block: src 192.168.1.2, dest 192.168.1.1
OAK_AG exchange ISAKMP (0): processing HASH payload. message ID = 0 ISAKMP (0): processing
NOTIFY payload 24578 protocol 1 spi 0, message ID = 0 ISAKMP (0): processing notify
INITIAL_CONTACT IPSEC(key_engine): got a queue event... IPSEC(key_engine_delete_sas): rec'd
delete notify from ISAKMP IPSEC(key_engine_delete_sas): delete all SAs shared with 192.168.1.2
ISAKMP (0): SA has been authenticated return status is IKMP_NO_ERROR ISAKMP/xauth: request
attribute XAUTH_TYPE ISAKMP/xauth: request attribute XAUTH_USER_NAME ISAKMP/xauth: request
attribute XAUTH_USER_PASSWORD ISAKMP (0:0): initiating peer config to 192.168.1.2. ID =
1623347510 (0x60c25136) crypto_isakmp_process_block: src 192.168.1.2, dest 192.168.1.1
ISAKMP_TRANSACTION exchange ISAKMP (0:0): processing transaction payload from 192.168.1.2.
message ID = 84 ISAKMP: Config payload CFG_REPLY return status is IKMP_ERR_NO_RETRANS ISAKMP
(0:0): initiating peer config to 192.168.1.2. ID = 2620656926 (0x9c340d1e)
```

```
crypto_isakmp_process_block: src 192.168.1.2, dest 192.168.1.1 ISAKMP_TRANSACTION exchange
ISAKMP (0:0): processing transaction payload from 192.168.1.2. message ID = 60 ISAKMP: Config
payload CFG_ACK return status is IKMP_NO_ERROR crypto_isakmp_process_block: src 192.168.1.2,
dest 192.168.1.1 ISAKMP_TRANSACTION exchange ISAKMP (0:0): processing transaction payload from
192.168.1.2. message ID = 0 ISAKMP: Config payload CFG_REQUEST ISAKMP (0:0): checking request:
ISAKMP: attribute IP4_ADDRESS (1) ISAKMP: attribute IP4_NETMASK (2) ISAKMP: attribute IP4_DNS
(3) ISAKMP: attribute IP4_NBNS (4) ISAKMP: attribute ADDRESS_EXPIRY (5) Unsupported Attr: 5
ISAKMP: attribute APPLICATION_VERSION (7) Unsupported Attr: 7 ISAKMP: attribute UNKNOWN (28672)
Unsupported Attr: 28672 ISAKMP: attribute UNKNOWN (28673) Unsupported Attr: 28673 ISAKMP:
attribute UNKNOWN (28674) ISAKMP: attribute UNKNOWN (28676) ISAKMP: attribute UNKNOWN (28679)
Unsupported Attr: 28679 ISAKMP: attribute UNKNOWN (28680) Unsupported Attr: 28680 ISAKMP:
attribute UNKNOWN (28677) Unsupported Attr: 28677 ISAKMP (0:0): responding to peer config from
192.168.1.2. ID = 177917346 return status is IKMP_NO_ERROR crypto_isakmp_process_block: src
192.168.1.2, dest 192.168.1.1 OAK_QM exchange oakley_process_quick_mode: OAK_QM_IDLE ISAKMP (0):
processing SA payload. message ID = 942875080 ISAKMP : Checking IPSec proposal 1 ISAKMP:
transform 1, ESP_3DES ISAKMP: attributes in transform: ISAKMP: authenticator is HMAC-MD5 ISAKMP:
encaps is 1 ISAKMP: SA life type in seconds ISAKMP: SA life duration (VPI) of 0x0 0x20 0xc4 0x9b
IPSEC(validate_proposal): transform proposal (prot 3, trans 3, hmac_alg 1) not supported ISAKMP
(0): atts not acceptable. Next payload is 0 ISAKMP (0): skipping next ANDed proposal (1) ISAKMP
: Checking IPSec proposal 2 ISAKMP: transform 1, ESP_3DES ISAKMP: attributes in transform:
ISAKMP: authenticator is HMAC-SHA ISAKMP: encaps is 1 ISAKMP: SA life type in seconds ISAKMP: SA
life duration (VPI) of 0x0 0x20 0xc4 0x9b IPSEC(validate_proposal): transform proposal (prot 3,
trans 3, hmac_alg 2) not supported ISAKMP (0): atts not acceptable. Next payload is 0 ISAKMP
(0): skipping next ANDed proposal (2) ISAKMP: Checking IPSec proposal 3 ISAKMP: transform 1,
ESP_3DES ISAKMP: attributes in transform: ISAKMP: authenticator is HMAC-MD5 ISAKMP: encaps is 1
ISAKMP: SA life type in seconds ISAKMP: SA life duration (VPI) of 0x0 0x20 0xc4 0x9b
IPSEC(validate_proposal): transform proposal (prot 3, trans 3, hmac_alg 1) not supported ISAKMP
(0): atts not acceptable. Next payload is 0 ISAKMP: Checking IPSec proposal 4 ISAKMP: transform
1, ESP_3DES ISAKMP: attributes in transform: ISAKMP: authenticator is HMAC-SHA ISAKMP: encaps is
1 ISAKMP: SA life type in seconds ISAKMP: SA life duration (VPI) of 0x0 0x20 0xc4 0x9b
IPSEC(validate_proposal): transform proposal (prot 3, trans 3, hmac_alg 2) not supported ISAKMP
(0): atts not acceptable. Next payload is 0 ISAKMP : Checking IPSec proposal 5 ISAKMP: transform
1, ESP_DES ISAKMP: attributes in transform: ISAKMP: authenticator is HMAC-MD5 ISAKMP: encaps is
1 ISAKMP: SA life type in seconds ISAKMP: SA life duration (VPI) of 0x0 0x20 0xc4 0x9b ISAKMP
(0): atts are acceptable. ISAKMP (0): bad SPI size of 2 octets! ISAKMP: Checking IPSec proposal
6 ISAKMP: transform 1, ESP_DES ISAKMP: attributes in transform: ISAKMP: authenticator is HMAC-
SHA ISAKMP: encaps is 1 ISAKMP: SA life type in seconds ISAKMP: SA life duration (VPI) of 0x0
0x20 0xc4 0x9b IPSEC(validate_proposal): transform proposal (prot 3, trans 2, hmac_alg 2) not
supported ISAKMP (0): atts not acceptable. Next payload is 0 ISAKMP (0): skipping next ANDed
proposal (6) ISAKMP : Checking IPSec proposal 7 ISAKMP: transform 1, ESP_DES ISAKMP: attributes
in transform: ISAKMP: authenticator is HMAC-MD5 ISAKMP: encaps is 1 ISAKMP: SA life type in
seconds ISAKMP: SA life duration (VPI) of 0x0 0x20 0xc4 0x9b ISAKMP (0): atts are
acceptable.IPSEC(validate_proposal_request): proposal part #1, (key eng. msg.) dest=
192.168.1.1, src= 192.168.1.2, dest_proxy= 192.168.1.1/255.255.255.255/0/0 (type=1), src_proxy=
10.89.129.200/255.255.255.255/0/0 (type=1), protocol= ESP, transform= esp-des esp-md5-hmac ,
lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 ISAKMP (0): processing
NONCE payload. message ID = 942875080 ISAKMP (0): processing ID payload. message ID = 942875080
ISAKMP (0): ID_IPV4_ADDR src 10.89.129.200 prot 0 port 0 ISAKMP (0): processing ID payload.
message ID = 942875080 ISAKMP (0): ID_IPV4_ADDR dst 192.168.1.1 prot 0 port 0IPSEC(key_engine):
got a queue event... IPSEC(spi_response): getting spi 0x64d7a518(1691854104) for SA from
192.168.1.2 to 192.168.1.1 for prot 3 return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.2, dest 192.168.1.1 OAK_QM exchange
oakley_process_quick_mode: OAK_QM_IDLE ISAKMP (0): processing SA payload. message ID =
3008609960 ISAKMP: Checking IPSec proposal 1 ISAKMP: transform 1, ESP_3DES ISAKMP: attributes in
transform: ISAKMP: authenticator is HMAC-MD5 crypto_isakmp_process_block: src 192.168.1.2, dest
192.168.1.1 OAK_QM exchange oakley_process_quick_mode: OAK_QM_AUTH_AWAITmap_alloc_entry:
allocating entry 2 map_alloc_entry: allocating entry 1 ISAKMP (0): Creating IPSec SAs inbound SA
from 192.168.1.2 to 192.168.1.1 (proxy 10.89.129.200 to 192.168.1.1) has spi 1691854104 and
conn_id 2 and flags 4 lifetime of 2147483 seconds outbound SA from 192.168.1.1 to 192.168.1.2
(proxy 192.168.1.1 to 10.89.129.200) has spi 1025193431 and conn_id 1 and flags 4 lifetime of
2147483 seconds IPSEC(key_engine): got a queue event... IPSEC(initialize_sas): ,(key eng. msg.)
dest= 192.168.1.1, src= 192.168.1.2, dest_proxy= 192.168.1.1/0.0.0.0/0/0 (type=1), src_proxy=
10.89.129.200/0.0.0.0/0/0 (type=1), protocol= ESP, transform= esp-des esp-md5-hmac , lifedur=
2147483s and 0kb, spi= 0x64d7a518(1691854104),conn_id= 2, keysize= 0, flags= 0x4
```

```
IPSEC(initialize_sas): , (key eng. msg.) src= 192.168.1.1, dest=192.168.1.2, src_proxy=
192.168.1.1/0.0.0.0/0/0 (type=1), dest_proxy= 10.89.129.200/0.0.0.0/0/0 (type=1), protocol= ESP,
transform=esp-des esp-md5-hmac , lifedur= 2147483s and 0kb, spi= 0x3d1b35d7(1025193431),conn_id=
1, keysize= 0, flags= 0x4 VPN Peer: IPSEC: Peer ip:192.168.1.2 Ref cnt incremented to:2 Total
VPN Peers:1 VPN Peer: IPSEC: Peer ip:192.168.1.2 Ref cnt incremented to:3 Total VPN Peers:1
return status is IKMP_NO_ERROR crypto_isakmp_process_block: src 192.168.1.2, dest 192.168.1.1
OAK_QM exchange oakley_process_quick_mode: OAK_QM_AUTH_AWAITmap_alloc_entry: allocating entry 4
map_alloc_entry: allocating entry 3 ISAKMP (0): Creating IPSec SAs inbound SA from 192.168.1.2
to 192.168.1.1 (proxy 10.89.129.200 to 0.0.0.0) has spi 3415657865 and conn_id 4 and flags 4
lifetime of 2147483 seconds outbound SA from 192.168.1.1 to 192.168.1.2 (proxy 0.0.0.0 to
10.89.129.200) has spi 2383969893 and conn_id 3 and flags 4 lifetime of 2147483
secondsIPSEC(key_engine): got a queue event... IPSEC(initialize_sas): , (key eng. msg.) dest=
192.168.1.1, src=192.168.1.2, dest_proxy= 0.0.0.0/0.0.0.0/0/0 (type=4), src_proxy=
10.89.129.200/0.0.0.0/0/0 (type=1), protocol= ESP, transform=esp-des esp-md5-hmac , lifedur=
2147483s and 0kb, spi= 0xcb96cd89(3415657865),conn_id= 4, keysize= 0, flags= 0x4
IPSEC(initialize_sas): , (key eng. msg.) src= 192.168.1.1, dest=192.168.1.2, src_proxy=
0.0.0.0/0.0.0.0/0/0 (type=4), dest_proxy= 10.89.129.200/0.0.0.0/0/0 (type=1), protocol= ESP,
transform=esp-des esp-md5-hmac , lifedur= 2147483s and 0kb, spi= 0x8e187e65(2383969893),conn_id=
3, keysize= 0, flags= 0x4 VPN Peer: IPSEC: Peer ip:192.168.1.2 Ref cnt incremented to:4 Total
VPN Peers:1 VPN Peer: IPSEC: Peer ip:192.168.1.2 Ref cnt incremented to:5 Total VPN Peers:1
return status is IKMP_NO_ERROR pixfirewall#show uauth
Current      Most Seen
Authenticated Users
1              1
Authen In Progress
0              1
ipsec user 'cisco_customer' at 10.89.129.200, authenticated
pixfirewall#
```

Débogues avec le client VPN 1.1

```
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
VPN Peer: ISAKMP: Added new peer: ip:192.168.1.3
Total VPN Peers:1
VPN Peer: ISAKMP: Peer ip:192.168.1.3 Ref cnt incremented to:1
Total VPN Peers:1
OAK_MM exchange
ISAKMP (0): processing SA payload. message ID = 0

ISAKMP (0): Checking ISAKMP transform 1 against priority 10 policy
      encryption DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 1
ISAKMP:      auth pre-share
ISAKMP (0): atts are not acceptable. Next payload is 0
ISAKMP (0): Checking ISAKMP transform 1 against priority 20 policy
ISAKMP:      encryption DES-CBC
ISAKMP:      hash MD5
ISAKMP:      default group 1
ISAKMP:      auth pre-share
ISAKMP (0): atts are acceptable. Next payload is 0
ISAKMP (0): SA is doing pre-shared key authentication
using id type ID_IPV4_ADDR
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
OAK_MM exchange
ISAKMP (0): processing KE payload. message ID = 0

ISAKMP (0): processing NONCE payload. message ID = 0

ISAKMP (0): processing vendor id payload

ISAKMP (0): processing vendor id payload
```

```
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
OAK_MM exchange
ISAKMP (0): processing ID payload. message ID = 0
ISAKMP (0): processing HASH payload. message ID = 0
ISAKMP (0): processing NOTIFY payload 24578 protocol 1
 spi 0, message ID = 0
ISAKMP (0): SA has been authenticated

ISAKMP (0): ID payload
 next-payload : 8
 type         : 1
 protocol     : 17
 port         : 500
 length       : 8
ISAKMP (0): Total payload length: 12
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
ISAKMP: Created a peer node for 192.168.1.3
OAK_QM exchange
ISAKMP (0:0): Need XAUTH
ISAKMP/xauth: request attribute XAUTH_TYPE
ISAKMP/xauth: request attribute XAUTH_USER_NAME
ISAKMP/xauth: request attribute XAUTH_USER_PASSWORD
ISAKMP (0:0): initiating peer config to 192.168.1.3.
ID = 3196940891 (0xbe8d725b)
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
ISAKMP_TRANSACTION exchange
ISAKMP (0:0): processing transaction payload
from 192.168.1.3. message ID = 84
ISAKMP: Config payload CFG_REPLY
return status is IKMP_ERR_NO_RETRANS
ISAKMP (0:0): initiating peer config to 192.168.1.3.
ID = 3196940891 (0xbe8d725b)
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
ISAKMP_TRANSACTION exchange
ISAKMP (0:0): processing transaction payload
from 192.168.1.3. message ID = 60
ISAKMP: Config payload CFG_ACK
ISAKMP (0:0): initiating peer config to 192.168.1.3.
ID = 1647424595 (0x6231b453)
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
ISAKMP_TRANSACTION exchange
ISAKMP (0:0): processing transaction payload
from 192.168.1.3. message ID = 60
ISAKMP: Config payload CFG_ACK
ISAKMP (0:0): peer accepted the address!
return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
OAK_QM exchange
oakley_process_quick_mode:
OAK_QM_IDLE
ISAKMP (0): processing SA payload. message ID = 802013669

ISAKMP : Checking IPSec proposal 1

ISAKMP: transform 1, ESP_DES
ISAKMP:   attributes in transform:
ISAKMP:      authenticator is HMAC-MD5
ISAKMP:      encaps is 1
ISAKMP (0): atts are acceptable.IPSEC(validate_proposal_request)
```

```
:proposal part #1,
  (key eng. msg.) dest= 192.168.1.1, src = 192.168.1.3,
    dest_proxy= 10.89.129.128/255.255.255.128/0/0 (type=4),
    src_proxy= 10.89.129.200/255.255.255.255/0/0 (type=1),
    protocol= ESP, transform=esp-des esp-md5-hmac ,
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize=0, flags= 0x4

ISAKMP (0): processing NONCE payload. message ID = 802013669

ISAKMP (0): processing ID payload. message ID = 802013669
ISAKMP (0): ID_IPV4_ADDR src 10.89.129.200 prot 0 port 0
ISAKMP (0): processing ID payload. message ID = 802013669
ISAKMP (0): ID_IPV4_ADDR_SUBNET dst 10.89.129.128/255.255.255.128
prot 0 port 0IPSEC(key_engine): got a queue event...
IPSEC(spi_response): getting spi 0xd7cef5ba(3620664762)for SA
 from 192.168.1.3 to 192.168.1.1 for prot 3

return status is IKMP_NO_ERROR
crypto_isakmp_process_block: src 192.168.1.3, dest 192.168.1.1
OAK_QM exchange
oakley_process_quick_mode:
OAK_QM_AUTH_AWAITmap_alloc_entry: allocating entry 1
map_alloc_entry: allocating entry 2

ISAKMP (0): Creating IPSec SAs
        inbound SA from 192.168.1.3 to 192.168.1.1
          (proxy 10.89.129.200 to 10.89.129.128)
        has spi 3620664762 and conn_id 1 and flags 4
        outbound SA from 192.168.1.1 to 192.168.1.3
          (proxy 10.89.129.128 to 10.89.129.200)
        has spi 541375266 and conn_id 2 and flags 4
IPSEC(key_engine): got a queue event...

IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 192.168.1.1, src=192.168.1.3,
    dest_proxy= 10.89.129.128/255.255.255.128/0/0 (type=4),
    src_proxy= 10.89.129.200/0.0.0.0/0/0 (type=1),
    protocol= ESP, transform=esp-des esp-md5-hmac ,
    lifedur= 0s and 0kb,
    spi= 0xd7cef5ba(3620664762),conn_id= 1, keysize= 0, flags= 0x4
IPSEC(initialize_sas): ,
  (key eng. msg.) src= 192.168.1.1, dest=192.168.1.3,
    src_proxy= 10.89.129.128/255.255.255.128/0/0 (type=4),
    dest_proxy= 10.89.129.200/0.0.0.0/0/0 (type=1),
    protocol= ESP, transform=esp-des esp-md5-hmac ,
    lifedur= 0s and 0kb,
    spi= 0x2044bb22(541375266),conn_id= 2, keysize= 0, flags= 0x4

VPN Peer: IPSEC: Peer ip:192.168.1.3 Ref cnt incremented
to:2 Total VPN Peers:1
VPN Peer: IPSEC: Peer ip:192.168.1.3 Ref cnt incremented
to:3 Total VPN Peers:1
return status is IKMP_NO_ERROR
```

Informations connexes

- Appliances de sécurité de la gamme PIX 500
- Références des commandes du pare-feu PIX
- Négociation IPSec/Protocoles IKE
- Introduction à IPSec

- Établissement de la connectivité via les pare-feu Cisco PIX
- Demandes de commentaires (RFC)
- Support et documentation techniques - Cisco Systems