Probleemoplossing voor de PIX om gegevensverkeer via een ingestelde IPSec-tunnelband door te geven

Inhoud

Inleiding

Dit document richt zich op en biedt een oplossing voor het probleem waarom een succesvolle IPsec-tunnel van een Cisco VPN-client naar een PIX geen gegevens kan doorgeven.

Het onvermogen om gegevens over een vastgestelde IPsec-tunnel tussen een VPN-client en een PIX door te geven wordt vaak aangetroffen wanneer u niet kunt pingelen of tellen van een VPN-client naar hosts op het netwerk achter de PIX. Met andere woorden, de VPN-client en PIX kunnen versleutelde gegevens niet tussen hen doorgeven. Dit komt voor omdat PIX een LAN-to-LAN IPsec tunnel aan een router en ook een VPN-client heeft. Het onvermogen om gegevens door te geven is het resultaat van een configuratie met dezelfde toegangscontrolelijst (ACL) voor zowel NAT 0 als de statische crypto kaart voor de LAN-to-LAN IPsec peer.

Voorwaarden

Vereisten

Er zijn geen specifieke vereisten van toepassing op dit document.

Gebruikte componenten

De informatie in dit document is gebaseerd op de volgende software- en hardware-versies:

- Cisco Secure PIX-firewall 6.0.1
- Cisco 1720 router die Cisco IOS®-softwarerelease 12.2(6)XR draait

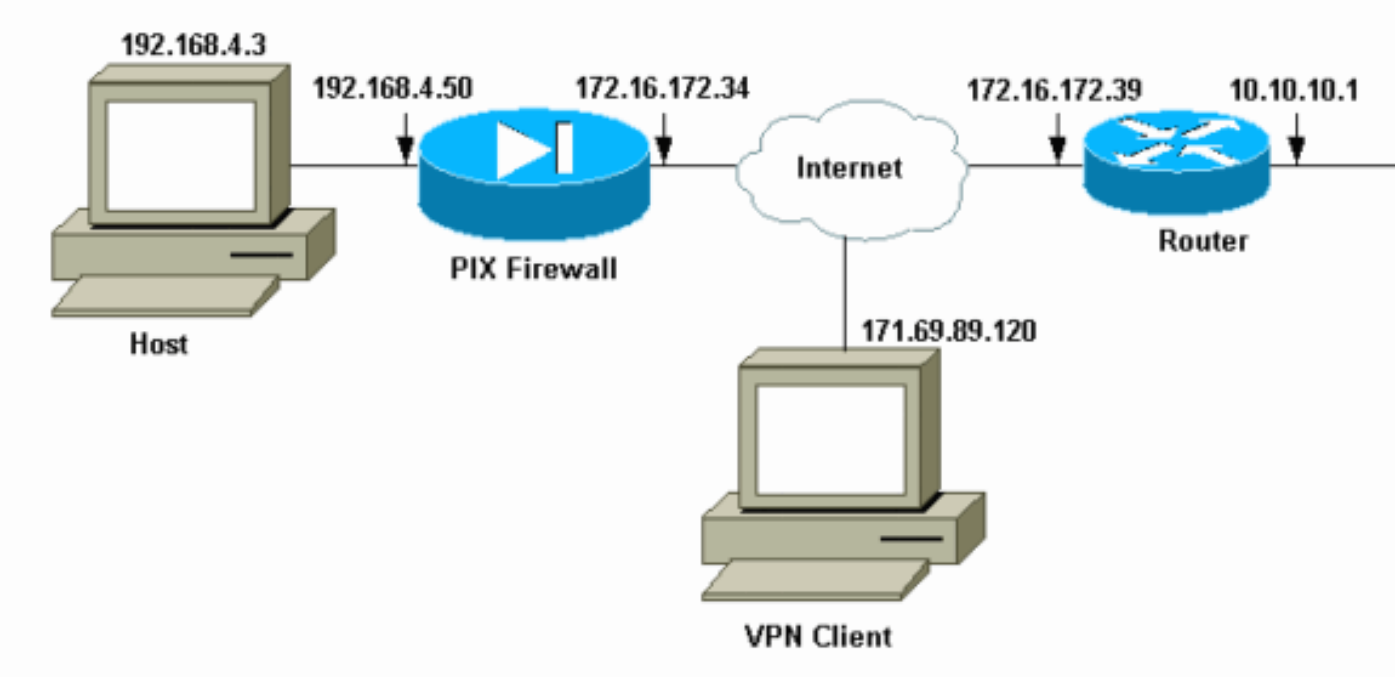
De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u de potentiële impact van elke opdracht begrijpen.

Conventies

Raadpleeg de Cisco Technical Tips Convention voor meer informatie over documentconventies.

Probleemoplossing voor PIX

Netwerkdiagram



Configuratie met problemen

PIX 520

```
pix520-1#write terminal
Building configuration...
: Saved
:
PIX Version 6.0(1)
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 2KFQnbNIdI.2KYOU encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pix520-1
domain-name vpn.com
fixup protocol ftp 21
```

```
fixup protocol http 80
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
names
!--- Access-List "140" defines interesting traffic to
bypass NAT for VPN !--- and defines VPN interesting
traffic. This is incorrect. access-list 140 permit ip
192.168.4.0 255.255.255.0 10.10.10.0 255.255.255.0
access-list 140 permit ip 192.168.4.0 255.255.255.0
10.1.2.0 255.255.255.0
no pager
logging on
logging console debugging
logging monitor debugging
logging buffered debugging
logging trap debugging
logging history debugging
logging host outside 192.168.2.6
interface ethernet0 auto
interface ethernet1 auto
mtu outside 1500
mtu inside 1500
!--- IP addresses on the outside and inside interfaces.
ip address outside 172.16.172.34 255.255.255.240
ip address inside 192.168.4.50 255.255.255.0
ip audit info action alarm
ip audit attack action alarm
ip local pool ippool 10.1.2.1-10.1.2.254
no failover
failover timeout 0:00:00
failover poll 15
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
pdm history enable
arp timeout 14400
global (outside) 1 172.16.172.57 netmask 255.255.255.255
!--- The nat 0 command bypasses NAT for the packets
destined over the IPsec tunnel.

Nat (inside) 0 access-list 140
Nat (inside) 1 0.0.0.0 0.0.0.0 0 0
route outside 0.0.0.0 0.0.0.0 172.16.172.33 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc
0:10:00 h323
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 mytest protocol tacacs+
AAA-server nasir protocol radius
snmp-server host outside 192.168.2.6
no snmp-server location
no snmp-server contact
snmp-server community public
snmp-server enable traps
floodguard enable
!--- The sysopt command bypasses conduits or ACLs that
check to be applied !--- on the inbound VPN packets
```

```
after decryption.

sysopt connection permit-ipsec
no sysopt route dnat
!--- The crypto ipsec command defines IPsec encryption
and authen algo.

crypto ipsec transform-set myset esp-des esp-md5-hmac
crypto dynamic-map dynmap 10 set transform-set myset
!--- The crypto map commands define the IPsec !---
Security Assocation (SA) (Phase II SA) parameters.

crypto map mymap 5 ipsec-isakmp
crypto map mymap 5 match address 140
crypto map mymap 5 set peer 172.16.172.39
crypto map mymap 5 set transform-set myset
crypto map mymap 10 ipsec-isakmp dynamic dynmap
crypto map mymap interface outside
isakmp enable outside
!--- The isakmp key command defines the pre-shared key
for the peer address.

isakmp key ******** address 172.16.172.39 netmask
255.255.255.255 no-xauth
no-config-mode
isakmp identity address
!--- The isakmp policy defines the Phase 1 SA
parameters.

isakmp policy 10 authentication pre-share
isakmp policy 10 encryption des
isakmp policy 10 hash sha
isakmp policy 10 group 2
isakmp policy 10 lifetime 86400
isakmp policy 20 authentication pre-share
isakmp policy 20 encryption Des
isakmp policy 20 hash sha
isakmp policy 20 group 1
isakmp policy 20 lifetime 86400
vpngroup vpn3000 address-pool ippool
vpngroup vpn3000 idle-time 1800
vpngroup vpn3000 password ********
telnet 192.168.4.0 255.255.255.0 inside
telnet 171.69.89.82 255.255.255.255 inside
telnet timeout 5
ssh 172.0.0.0 255.0.0.0 outside
ssh 171.0.0.0 255.255.255.0 outside
ssh 171.0.0.0 255.0.0.0 outside
ssh timeout 60
terminal width 80
Cryptochecksum:55948dc706cc700e9c10e1d24a8b125c
```

In de problematische configuratie wordt het interessante verkeer, of het verkeer dat moet worden versleuteld voor de LAN-to-LAN tunnel, gedefinieerd door ACL 140. De configuratie gebruikt dezelfde ACL als de NAT 0 ACL.

Begrijp de algemene volgorde van gebeurtenissen

Wanneer een IP-pakket wordt ontvangen op de interne interface van de PIX, wordt de Netwerkadresomzetting (NAT) gecontroleerd. Daarna worden ACL's voor crypto kaarten gecontroleerd.

- **Hoe wordt NAT 0 gebruikt.**Het NAT 0 ACL definieert wat niet in NAT moet worden opgenomen. Het ACL in het **NAT 0** bevel definieert het bron- en doeladres waarvoor de NAT regels over de PIX worden uitgeschakeld. Daarom omzeilt een IP-pakket met een bron- en doeladres dat overeenkomt met de ACL die in het **NAT 0** is gedefinieerd, alle NAT-regels op de PIX.Om LAN-to-LAN tunnels tussen een PIX en een ander VPN-apparaat te implementeren met de hulp van de privé-adressen, gebruikt u de **NAT 0-**opdracht om NAT te omzeilen. De regels in de PIX-firewall verhinderen dat de privé-adressen in NAT worden opgenomen terwijl deze regels naar het externe LAN via de IPsec-tunnel gaan.
- **Hoe crypto ACL wordt gebruikt.**Na de NAT-inspecties, controleert de PIX de bron en de bestemming van elk IP-pakket dat op zijn interne interface arriveert om de ACL's aan te passen die in de statische en dynamische crypto-kaarten zijn gedefinieerd. Als PIX een overeenkomst met ACL vindt, neemt PIX een van deze stappen:Als er geen huidige IPsec Security Association (SA) bestaat die al met het peer IPsec-apparaat voor het verkeer is gebouwd, start PIX de IPsec-onderhandelingen. Zodra de SAs worden gebouwd, versleutelt het pakket en stuurt het via de IPsec-tunnel naar de IPsec-peer.Als er al een IPsec SA is gebouwd met de peer, versleutelt PIX het IP-pakket en stuurt het gecodeerde pakket naar het IPsec-apparaat.
- **Dynamische ACL.**Zodra een VPN-client met de hulp van IPsec op de PIX is aangesloten, maakt de PIX een dynamische ACL die het bron- en doeladres specificeert om het interessante verkeer voor deze IPsec-verbinding te definiëren.

Heb begrip voor de problematische serie gebeurtenissen in de PIX

Een algemene configuratiefout is om dezelfde ACL te gebruiken voor NAT 0 en de statische crypto kaarten. In deze paragrafen wordt besproken waarom dit tot een fout leidt en hoe het probleem kan worden opgelost.

De configuratie van PIX toont aan dat het nummer 0 ACL 140 NAT passeert wanneer IP-pakketten van netwerk 192.168.4.0/24 naar netwerken 10.10.10.0/24 en 10.1.2.0/24 (netwerkadres gedefinieerd in de IP lokale pool) gaan. Daarnaast definieert ACL 140 het interessante verkeer voor de statische cryptokaart voor peer 172.16.172.39.

Wanneer een IP-pakket naar de PIX-binneninterface komt, wordt de NAT-controle voltooid en vervolgens controleert de PIX de ACL's in de crypto-kaarten. De PIX begint met de crypto kaart met het laagste nummer. Dit is omdat de statische crypto kaart in het vorige voorbeeld het laagste aantal voorbeelden heeft, wordt ACL 140 gecontroleerd. Daarna wordt de dynamische ACL voor de dynamische crypto map gecontroleerd. In deze configuratie is ACL 140 gedefinieerd om verkeer te versleutelen dat van netwerk 192.168.4.0/24 naar netwerken 10.10.10.0/24 0 en 10.1.2.0/24 gaat. Voor de LAN-to-LAN tunnel wilt u echter alleen verkeer versleutelen tussen netwerken 192.168.4.0/2 4 en 10.10.10.0/24. Dit is hoe de IPsec peer router zijn crypto ACL definieert.

Heb begrip voor de problematische serie gebeurtenissen in de PIX

Wanneer een client een IPsec-verbinding naar de PIX maakt, krijgt deze een IP-adres toegewezen in de lokale IP-pool. In dit geval wordt de client toegewezen 10.1.2.1. De PIX genereert ook een dynamische ACL, zoals deze opdrachtoutput van **crypto map** toont:

```
Crypto Map "mymap" 20 ipsec-isakmp
```

```
Peer = 171.69.89.120
access-list dynacl2 permit ip host 172.16.172.34 host 10.1.2.1 (hitcnt=0)
dynamic (created from dynamic map dynmap/10)
Current peer: 171.69.89.120
Security association lifetime: 4608000 kilobytes/28800 seconds
PFS (Y/N): N
Transform sets={ myset, }
Crypto Map "mymap" 30 ipsec-isakmp
Peer = 171.69.89.120
access-list dynacl3 permit ip any host 10.1.2.1 (hitcnt=0)
dynamic (created from dynamic map dynmap/10)
Current peer: 171.69.89.120
Security association lifetime: 4608000 kilobytes/28800 seconds
PFS (Y/N): N
Transform sets={ myset, }
pix520-1(config)#
```

De opdracht **show crypto map** toont ook de statische crypto kaart:

```
Crypto Map: "mymap" interfaces: { outside }
Crypto Map "mymap" 5 ipsec-isakmp
Peer = 172.16.172.39
access-list 140 permit ip 192.168.4.0 255.255.255.0 10.10.10.0255.255.255.0
   (hitcnt=45)
access-list 140 permit ip 192.168.4.0 255.255.255.0 10.1.2.0 255.255.255.0
   (hitcnt=84)
Current peer: 172.16.172.39
Security association lifetime: 4608000 kilobytes/28800 seconds
PFS (Y/N): N
Transform sets={ myset,}
```

Zodra de IPsec-tunnel tussen de client en de PIX tot stand is gebracht, initieert de client een ping naar de host 192.168.4.3. Wanneer de client het echo-verzoek ontvangt, reageert de host 192.168.4.3 met een echo-antwoord zoals deze uitvoer van de opdracht **debug icmp-sporen** aantoont.

```
27: Inbound ICMP echo request (len 32 id 2 seq 7680)
   10.1.2.1 > 192.168.4.3> 192.168.4.3
28: Outbound ICMP echo reply (Len 32 id 2 seq 7680)
   192.168.4.3 >192.168.4.3 > 10.1.2.1
29: Inbound ICMP echo request (Len 32 id 2 seq 7936)
   10.1.2.1 > 192.168.4.3> 192.168.4.3
30: Outbound ICMP echo reply (Len 32 id 2 seq 7936)
   192.168.4.3 >192.168.4.3 > 10.1.2.1
```

Het echo-antwoord bereikt echter niet de VPN-client (host 10.1.2.1) en ping mislukt. U kunt dit zien met behulp van de **show crypto ipsec als** opdracht op de PIX. Deze uitvoer toont dat PIX 120 pakketten decrypteert die van de client van VPN komen, maar het versleutelt geen pakketten of de gecodeerde pakketten naar de client. Daarom is het aantal ingekapselde pakketten nul.

```
pix520-1(config)#show crypto ipsec sa
interface: outside
Crypto map tag: mymap, local addr. 172.16.172.34
local ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
remote ident (addr/mask/prot/port): (10.1.2.1/255.255.255.255/0/0)
current_peer: 171.69.89.120
dynamic allocated peer ip: 10.1.2.1
PERMIT, flags={}
```

```
#pkts encaps: 0, #pkts encrypt: 0, #pkts digest 0
!--- No packets encrypted and sent to client. #pkts decaps: 120, #pkts decrypt: 120, #pkts
verify 120
!--- 120 packets received from client. #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.: 172.16.172.34, remote crypto endpt.: 171.69.89.120
path mtu 1500, ipsec overhead 56, media mtu 1500
current outbound spi: 33a45029
inbound esp sas:
spi: 0x279fc5e9(664782313)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 5, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4607985/27809)
IV size: 8 bytes
replay detection support: Y
inbound ah sas:
inbound pcp sas:
outbound ESP sas:
spi: 0x33a45029(866406441)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 6, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4608000/27809)
IV size: 8 bytes
replay detection support: Y
outbound ah sas:
outbound PCP sas:
local ident (addr/mask/prot/port): (192.168.4.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (10.10.10.0/255.255.255.0/0/0)
current_peer: 172.16.172.39
PERMIT, flags={origin_is_acl,}
#pkts encaps: 10, #pkts encrypt: 10, #pkts digest 10
#pkts decaps: 23, #pkts decrypt: 23, #pkts verify 23
#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.: 172.16.172.34, remote crypto endpt.: 172.16.172.39
path mtu 1500, ipsec overhead 56, media mtu 1500
current outbound spi: f264e92c
inbound ESP sas:
spi: 0x2772b869(661829737)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 1, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4607997/2420)
IV size: 8 bytes
replay detection support: Y
inbound ah sas:
inbound PCP sas:
outbound ESP sas:
spi: 0xf264e92c(4066699564)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 2, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4607999/2420)
IV size: 8 bytes
replay detection support: Y
outbound ah sas:
outbound PCP sas:
```

Opmerking: wanneer de host 192.168.4.3 reageert op het echo-verzoek, wordt het IP-pakket afgeleverd op de interne interface van de PIX.

```
38: Outbound ICMP echo reply (Len 32 id 2 seq 8960)
   192.168.4.3 >192.168.4.3 > 10.1.2.1
```

Zodra het IP-pakket op de interne interface arriveert, controleert PIX het nummer 0 ACL 140 en bepaalt hij dat de bron- en doeladressen van het IP-pakket overeenkomen met de ACL-code. Daarom passeert dit IP-pakket alle NAT-regels voor de PIX-map. Daarna worden de crypto ACL's gecontroleerd. Aangezien de statische crypto kaart het laagste aantal van de instantie heeft, wordt zijn ACL eerst gecontroleerd. Aangezien dit voorbeeld ACL 140 voor de statische crypto kaart gebruikt, controleert PIX dit ACL. Nu heeft het IP-pakket een bronadres van 192.168.4.3 en een bestemming van 10.1.2.1. Aangezien dit ACL 140 aanpast, denkt PIX dat dit IP-pakket is bedoeld voor de LAN-to-LAN IPsec-tunnel met peer 172.16.172.39 (in strijd met onze doelstellingen). Daarom controleert zij de SA-database om te zien of er al een huidige SA met peer 172.16.72.39 is voor dit verkeer. Zoals de uitvoer van de **show crypto ipsec als** opdracht aantoont, bestaat er geen SA voor dit verkeer. De PIX versleutelt het pakket niet of stuurt het niet naar de VPN-client. In plaats daarvan start het een andere IPsec-onderhandeling met peer 172.16.172.39 zoals deze uitvoer toont:

```
crypto_isakmp_process_block: src 172.16.172.39, dest 172.16.172.34
return status is IKMP_NO_ERR_NO_TRANS02303: sa_request, (key eng. msg.)
src= 172.16.172.34, dest= 172.16.172.39,
src_proxy= 192.168.4.0/255.255.255.0/0/0 (type=4),
dest_proxy= 10.1.2.0/255.255.255.0/0/0 (type=4), protocol= ESP, transform=
ESP-Des esp-md5-hmac , lifedur= 28800s and 4608000kb,
spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4004
702303: sa_request, (key Eng. msg.) src= 172.16.172.34, dest=
172.16.172.39, src_proxy= 192.168.4.0/255.255.255.0/0/0 (type=4),
dest_proxy= 10.1.2.0/255.255.255.0/0/0 (type=4), protocol= ESP, transform=
ESP-Des esp-md5-hmac , lifedur= 28800s and 4608000kb,
spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4004
ISAKMP (0): sending NOTIFY message 36137 protocol 1
return status is IKMP_NO_ERR_NO_TRANSIPSEC(key_engine): request timer
fired: count = 2,
(identity) local= 172.16.172.34, remote= 172.16.172.39,
local_proxy= 192.168.4.0/255.255.255.0/0/0 (type=4),
remote_proxy= 10.1.2.0/255.255.255.0/0/0 (type=4)
```

De IPsec-onderhandeling mislukt om deze redenen:

- De peer 172.16.172.39 definieert alleen netwerken 10.10.10.0/24 en 192.168.4.0/24 als het interessante verkeer in zijn ACL voor de crypto kaart peer 172.16.172.34.
- De proxy-identiteiten komen niet overeen tijdens de IPsec-onderhandeling tussen de twee peers.
- Als de peer de onderhandeling initieert en de lokale configuratie de perfecte voorwaartse geheimhouding (PFS) specificeert, moet de peer een PFS-uitwisseling uitvoeren of de onderhandeling mislukt. Als de lokale configuratie geen groep specificeert, wordt een standaard van groep1 aangenomen en wordt een aanbod van groep1 of groep2 geaccepteerd. Als de lokale configuratie group2 specificeert, moet die groep deel uitmaken van het aanbod van de peer of zal de onderhandeling mislukken. Als de lokale configuratie geen PFS specificeert, accepteert het elk aanbod van PFS van de peer. De 1024-bits Diffie-Hellman prime modulus groep, groep2, biedt meer beveiliging dan groep1, maar vereist meer verwerkingstijd dan groep1.**Opmerking:** de **crypto map set pfs** opdracht stelt IPsec in om naar PFS te vragen wanneer er nieuwe SA's worden gevraagd voor deze crypto map entry. Gebruik de **opdracht geen crypto map set pfs** om aan te geven dat IPsec niet om PFS verzoekt. Deze opdracht is alleen beschikbaar voor crypto-map-items van IPsec-ISAKMP en

dynamische crypto-map-items. Standaard wordt PFS niet gevraagd. Met PFS, elke keer als er een nieuwe SA wordt onderhandeld, vindt er een nieuwe Diffie-Hellman uitwisseling plaats. Dit vereist extra verwerkingstijd. PFS voegt een ander beveiligingsniveau toe omdat als een toets ooit door een aanvaller wordt gekraakt, alleen de gegevens die met die toets worden verstuurd, worden gecompromitteerd. Tijdens onderhandeling, veroorzaakt deze opdracht IPsec om PFS te vragen wanneer het nieuwe SAs voor de crypto kaartingang verzoekt. De default (group1) wordt verzonden als de **set pfs** statement geen groep specificeert.**Opmerking:** IKE onderhandelingen met een externe peer kunnen hangen wanneer een PIX-firewall talloze tunnels heeft die afkomstig zijn van de PIX-firewall en eindigen op één externe peer. Dit probleem doet zich voor als PFS niet is ingeschakeld en de lokale peer vraagt veel gelijktijdige rekey verzoeken. Als dit probleem zich voordoet, kan IKE SA niet herstellen tot het tijd uit is of tot u het handmatig wist met de **duidelijke [crypto] isakmp als** opdracht. PIX-firewalleenheden die met veel tunnels zijn geconfigureerd voor veel peers of veel klanten die dezelfde tunnel delen, worden niet door dit probleem getroffen. Als uw configuratie wordt beïnvloed, schakelt u PFS in met de opdracht **crypto kaart mapnaam** en **pfs-set**.

De IP-pakketten op de PIX worden uiteindelijk verbroken.

De oplossing begrijpen

De juiste methode om deze fout te corrigeren is om twee afzonderlijke ACL's voor NAT 0 en de statische crypto-kaarten te definiëren. Om dit te doen, definieert het voorbeeld ACL 190 voor het **anti 0** bevel en gebruikt het gewijzigde ACL 140 voor de statische crypto kaart, zoals deze uitvoer toont.

PIX 520-1

```
pix520-1(config)#
pix520-1(config)#write terminal
Building configuration...
: Saved
:
PIX Version 6.0(1)
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password 2KFQnbNIdI.2KYOU encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pix520-1
domain-name vpn.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
names
!--- Access list 140 defines interesting traffic in
order to bypass NAT for VPN. access-list 140 permit ip
192.168.4.0 255.255.255.0 10.10.10.0255.255.255.0
!--- Defines VPN interesting traffic. access-list 190
permit ip 192.168.4.0 255.255.255.0
10.10.10.0255.255.255.0
access-list 190 permit ip 192.168.4.0 255.255.255.0
10.1.2.0 255.255.255.0
```

```
no pager
logging on
logging console debugging
logging monitor debugging
logging buffered debugging
logging trap debugging

logging history debugging
logging host outside 192.168.2.6
interface ethernet0 auto
interface ethernet1 auto
mtu outside 1500
mtu inside 1500
ip address outside 172.16.172.34 255.255.255.240
ip address inside 192.168.4.50 255.255.255.0
ip audit info action alarm
ip audit attack action alarm
ip local pool ippool 10.1.2.1-10.1.2.254
no failover
failover timeout 0:00:00
failover poll 15
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
pdm history enable
arp timeout 14400
global (outside) 1 172.16.172.57 netmask 255.255.255.255
!--- The nat 0 command bypasses NAT for the packets
destined over the IPsec tunnel..

Nat (inside) 0 access-list 190
Nat (inside) 1 0.0.0.0 0.0.0.0 0 0
route outside 0.0.0.0 0.0.0.0 172.16.172.33 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc
0:10:00 h323
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 mytest protocol tacacs+
AAA-server nasir protocol radius
snmp-server host outside 192.168.2.6
no snmp-server location
no snmp-server contact
snmp-server community public
snmp-server enable traps
floodguard enable
sysopt connection permit-ipsec
no sysopt route dnat
crypto ipsec transform-set myset ESP-Des esp-md5-hmac
crypto dynamic-map dynmap 10 set transform-set myset
!--- The crypto map commands define the IPsec SA (Phase
II SA) parameters.

crypto map mymap 5 ipsec-isakmp
crypto map mymap 5 match address 140
crypto map mymap 5 set peer 172.16.172.39
crypto map mymap 5 set transform-set myset
crypto map mymap 10 ipsec-isakmp dynamic dynmap
crypto map mymap interface outside
isakmp enable outside
isakmp key ******** address 172.16.172.39 netmask
255.255.255.255 no-xauth
no-config-mode
```

```
isakmp identity address
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption Des
isakmp policy 10 hash sha
isakmp policy 10 group 2
isakmp policy 10 lifetime 86400
isakmp policy 20 authentication pre-share
isakmp policy 20 encryption Des
isakmp policy 20 hash sha
isakmp policy 20 group 1
isakmp policy 20 lifetime 86400
vpngroup vpn3000 address-pool ippool
vpngroup vpn3000 idle-time 1800
vpngroup vpn3000 password ********
telnet 192.168.4.0 255.255.255.0 inside
telnet 171.69.89.82 255.255.255.255 inside
telnet timeout 5
ssh 172.0.0.0 255.0.0.0 outside
ssh 171.0.0.0 255.255.255.0 outside
ssh 171.0.0.0 255.0.0.0 outside
ssh timeout 60
terminal width 80
Cryptochecksum:e2cb98b30d3899597b3af484fae4f9ae
: end
[OK]
pix520-1(config)# pix520-1(config)#show crypto map
```

Nadat de veranderingen worden aangebracht en de client een IPsec-tunnel met de PIX vastlegt, geeft u de opdracht tonen crypto kaart uit. Deze opdracht laat zien dat voor de statische crypto-kaart, het interessante verkeer dat door ACL 140 wordt gedefinieerd slechts 192.168.4.0/24 en 10.10.10.0/24 is, wat de oorspronkelijke doelstelling was. Daarnaast toont de dynamische toegangslijst het interessante verkeer dat gedefinieerd is als de cliënt (10.1.2.1) en de PIX (172.16.172.34).

```
pix520-1(config)#show crypto map
Crypto Map: "mymap" interfaces: { outside }
Crypto Map "mymap" 5 ipsec-isakmp
Peer = 172.16.172.39
access-list 140 permit ip 192.168.4.0 255.255.255.0 10.10.10.0255.255.255.0
   (hitcnt=57)
Current peer: 172.16.172.39
Security association lifetime: 4608000 kilobytes/28800 seconds
PFS (Y/N): N
Transform sets={ myset, }
Crypto Map "mymap" 10 ipsec-isakmp
Dynamic map template tag: dynmap
Crypto Map "mymap" 20 ipsec-isakmp
Peer = 171.69.89.120
access-list dynacl4 permit ip host 172.16.172.34 host 10.1.2.1 (hitcnt=0)
dynamic (created from dynamic map dynmap/10)
Current peer: 171.69.89.120
Security association lifetime: 4608000 kilobytes/28800 seconds
PFS (Y/N): N
Transform sets={ myset, }
Crypto Map "mymap" 30 ipsec-isakmp
Peer = 171.69.89.120
access-list dynacl5 permit ip any host 10.1.2.1 (hitcnt=13)
dynamic (created from dynamic map dynmap/10)
Current peer: 171.69.89.120
Security association lifetime: 4608000 kilobytes/28800 seconds
PFS (Y/N): N
```

```
Transform sets={ myset, }
```

Wanneer VPN-client 10.1.2.1 een ping naar host-192.168.4.3 stuurt, wordt het echo-antwoord naar de interne interface van de PIX verzonden. De PIX controleert het getal 0 ACL 190 en bepaalt dat het IP-pakket overeenkomt met ACL. Daarom passeert het pakje de NAT-regels voor de PIX. Daarna controleert de PIX de statische crypto kaart ACL 140 om een match te vinden. Dit keer komen de bron en de bestemming van het IP-pakket niet overeen met ACL 140. Daarom controleert PIX de dynamische ACL en vindt er een overeenkomst. PIX controleert vervolgens zijn SA-database om te zien of al dan niet een IPsec SA al met de client is ingesteld. Aangezien de client al een IPsec-verbinding met de PIX heeft gelegd, bestaat er een IPsec SA. De PIX versleutelt vervolgens de pakketten en stuurt het naar de VPN-client. Gebruik de opdrachtoutput van **show crypto ipsec als** opdracht van de PIX om te zien dat pakketten versleuteld en ontsleuteld zijn. In dit geval, versleutelde de PIX zestien pakketten en stuurde ze naar de client. PIX heeft ook versleutelde pakketten van de VPN-client ontvangen en zestien pakketten gedecrypteerd.

```
pix520-1(config)#show crypto ipsec sa
interface: outside
Crypto map tag: mymap, local addr. 172.16.172.34
local ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
remote ident (addr/mask/prot/port): (10.1.2.1/255.255.255.255/0/0)
current_peer: 171.69.89.120
dynamic allocated peer ip: 10.1.2.1
PERMIT, flags={}
#pkts encaps: 16, #pkts encrypt: 16,#pkts digest 16
#pkts decaps: 16, #pkts decrypt: 16, #pkts verify 16
#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.: 172.16.172.34, remote crypto endpt.: 171.69.89.120
path mtu 1500, ipsec overhead 56, media mtu 1500
current outbound spi: 613d083d
inbound ESP sas:
spi: 0x6adf97df(1793038303)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 4, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4607998/27420)
IV size: 8 bytes
replay detection support: Y
inbound ah sas:
inbound PCP sas:
outbound ESP sas:
spi: 0x613d083d(1631389757)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 3, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4607999/27420)
IV size: 8 bytes
replay detection support: Y
outbound ah sas:
outbound PCP sas:
local ident (addr/mask/prot/port): (192.168.4.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (10.10.10.0/255.255.255.0/0/0)
current_peer: 172.16.172.39
PERMIT, flags={origin_is_acl,}
#pkts encaps: 9, #pkts encrypt: 9, #pkts digest 9
#pkts decaps: 9, #pkts decrypt: 9, #pkts verify 9
#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.: 172.16.172.34, remote crypto endpt.: 172.16.172.39
```

```
path mtu 1500, ipsec overhead 56, media mtu 1500
current outbound spi: 58009c01
inbound ESP sas:
spi: 0x2d408709(759203593)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 2, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4607998/3319)
IV size: 8 bytes
replay detection support: Y
inbound ah sas:
inbound PCP sas: outbound ESP sas:
spi: 0x58009c01(1476434945)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 1, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4607999/3319)
IV size: 8 bytes
replay detection support: Y
outbound ah sas:
outbound PCP sas:
pix520-1(config)# sh cr isa sa
Total : 2
Embryonic : 0
dst src state pending created
172.16.172.39 172.16.172.34 QM_IDLE 0 1
172.16.172.34 171.69.89.120 QM_IDLE 0 2
pix520-1(config)# sh cr ipsec sa
```

Routerconfiguratie en -uitvoer tonen

Cisco 1720-1 switch

```
1720-1#show run
Building configuration...
Current configuration : 1592 bytes
!
! Last configuration change at 21:08:49 PST Mon Jan 7
2002
! NVRAM config last updated at 18:18:17 PST Mon Jan 7
2002
!
version 12.2
no parser cache
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname 1720-1
!
no logging buffered
enable secret 5 $1$6jAs$tNxI1a/2DYFAtPLyCDXjo/
enable password ww
!
username cisco password 0 cisco
memory-size iomem 15
clock timezone PST -8
ip subnet-zero
no ip domain-lookup
ip domain-name cisco.com
!
ip ssh time-out 120
```

```
ip ssh authentication-retries 3
!
!
!--- The crypto isakmp policy command defines the Phase
1 SA parameters.

crypto isakmp policy 15
authentication pre-share
crypto isakmp key cisco123 address 172.16.172.34
!
!
!--- The crypto ipsec transform-set command defines
IPsec encryption !--- and authentication algorithims.

crypto ipsec transform-set myset ESP-Des esp-md5-hmac
!
!
!--- The crypto map command defines the IPsec SA (Phase
II SA) parameters..

crypto map vpn 10 ipsec-isakmp
set peer 172.16.172.34
set transform-set myset
match address 150
!
!
!
!
!
interface FastEthernet0
ip address 172.16.172.39 255.255.255.240
speed auto
!--- The crypto map applied to the outbound interface.
crypto map vpn
interface Ethernet0
ip address 10.10.10.1 255.255.255.240
speed auto
no ip route-cache
no ip mroute-cache
!
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.16.172.33
no ip http server
ip pim bidir-enable
!
!--- Access-list defines interesting VPN traffic.
access-list 150 permit ip 10.10.10.0 0.0.0.255
192.168.4.0 0.0.0.255
!
line con 0
line aux 0
line vty 0 4
exec-timeout 0 0
password cisco
no login
line vty 5 15
login
!
no scheduler allocate
end
1720-1#
```

```
1720-1#show crypto isa sa
DST src state conn-id slot
172.16.172.39 172.16.172.34 QM_IDLE 132 0
1720-1#show crypto ipsec sa
interface: FastEthernet0
Crypto map tag: vpn, local addr. 172.16.172.39
local ident (addr/mask/prot/port): (10.10.10.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (192.168.4.0/255.255.255.0/0/0)
current_peer: 172.16.172.34
PERMIT, flags={origin_is_acl,}
#pkts encaps: 9 #pkts encrypt: 9 #pkts digest 9
#pkts decaps: 9, #pkts decrypt: 9, #pkts verify 9
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. Failed: 0, #pkts decompress failed: 0
#send errors 7, #recv errors 0
local crypto endpt.: 172.16.172.39, remote crypto endpt.: 172.16.172.34
path mtu 1500, media mtu 1500
current outbound spi: 2D408709
inbound ESP sas:
spi: 0x58009C01(1476434945)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
!--- IPsec SA 200 as seen in the show crypto engine connection active command.

slot: 0, conn id: 200, flow_id: 1, crypto map: vpn
sa timing: remaining key lifetime (k/sec): (4607998/3144)
IV size: 8 bytes
replay detection support: Y
inbound ah sas:
inbound PCP sas:
outbound ESP sas:
spi: 0x2D408709(759203593)
transform: ESP-Des esp-md5-hmac ,
in use settings ={Tunnel, }
!--- IPsec SA 201 as seen in the show crypto engine connection active command.

slot: 0, conn id: 201, flow_id: 2, crypto map: vpn
sa timing: remaining key lifetime (k/sec): (4607998/3144)
IV size: 8 bytes
replay detection support: Y
outbound ah sas:
outbound PCP sas:
1720-1#

1720-1#show crypto map
Interfaces using crypto map mymap:
Crypto Map "vpn" 10 ipsec-isakmp
Peer = 172.16.172.34
Extended IP access list 150
access-list 150 permit ip 10.10.10.0 0.0.0.255 192.168.4.0 0.0.0.255
Current peer: 172.16.172.34
Security association lifetime: 4608000 kilobytes/3600 seconds
PFS (Y/N): N
Transform sets={ myset, }
Interfaces using crypto map vpn: FastEthernet0
```

Gerelateerde informatie

- Cisco PIX-firewallsoftware
- Opdrachtreferenties van Cisco Secure PIX-firewall
- Security meldingen uit het veld (inclusief PIX)

- Verzoeken om opmerkingen (RFC's)
- IPsec-onderhandeling/IKE-protocollen
- Technische ondersteuning en documentatie – Cisco Systems