VPN Debugging Notes

Disclaimer: I do not work for Cisco or Checkpoint. This page is not supported, endorsed or approved by Checkpoint, Cisco, Nortel, Nokia, nor my employer. I do not hold myself out as an expert. This page is no more than a personal cheatsheet, put online to share with some colleagues. It isn't even spell-checked, and I'm a poor typist. This information is presented on an "as-is" basis without warranty of any kind. Use at your own risk. Since people have started arriving here now and then by Googling error messages, I've cleaned this up a little, but stranger, beware: these are still ROUGH NOTES whose instructions and advice are intended to apply to specific environments and they may or may not be appropriate for yours.

Quick Jumps

1. [Terminology](http://www.rick-thompson.com/misc/vpn-notes.html#terminology)
2. [Commonly seen symptoms and likely causes](http://www.rick-thompson.com/misc/vpn-notes.html#symptoms) 
   * [You're using a Checkpoint 4.1 box](http://www.rick-thompson.com/misc/vpn-notes.html#CP41)
   * [You're using a Checkpoint NG Box](http://www.rick-thompson.com/misc/vpn-notes.html#CPNG)
   * [You're using a Nortel](http://www.rick-thompson.com/misc/vpn-notes.html#Nortel)
   * [You're using a Cisco Box](http://www.rick-thompson.com/misc/vpn-notes.html#cisco)
   * [You're using an off-brand box](http://www.rick-thompson.com/misc/vpn-notes.html#otherbrands)
3. [Some Handy PIX / IOS syntax reminders](http://www.rick-thompson.com/misc/vpn-notes.html#syntax)
4. [A sample PIX config](http://www.rick-thompson.com/misc/vpn-notes.html#pixsample)
5. [Common mistakes in no particular order](http://www.rick-thompson.com/misc/vpn-notes.html#mistakes)
6. [Out into the weeds](http://www.rick-thompson.com/misc/vpn-notes.html#weeds)

Terminology

Phase 1 = ISAKMP  = IKE = "Main mode" = Oakley = SA's -- the IKE nodes negotiate and establish a secure link that will be used for the phase two. These are the Checkpoint properties of the gateway objects and the PIX policy definitions. The "preshared key" is an ISAKMP key, so if phase 1 completes, the key is OK -- The IPSec keys are created on the fly

Phase 2 = IPSec = "Quick mode" = SPI's  -- In phase two, the real negotiations are done. The IPsec SA is created. These are the Checkpoint properties of the encrypt action and the PIX transform-sets

Commonly seen symptoms and likely causes

You're using a Checkpoint 4.1 box

|  |  |  |
| --- | --- | --- |
| Platform | Symptom/Message | Likely cause or solution |
| Checkpoint 4.1  Why aren't you moving to NG? | Checkpoint log message of:  No proposal chosen | The most common failure symptom I've seen.  Sadly, a number of things can cause this. This is a failure in phase 1 -- it never gets to the point where it tries to process the "encrypt" action in the rule base, so the problem almost certainly lies in your definition of the peer gateway, or his definition of you. You and s/he can't agree on parameters or subnets for the initial negotiation.  Problems I've seen cause this, In order of likelihood:   * Mismatch in encryption algorithm (DES/3DES, etc), or hash method (SHA/MD5) on peer gateway object's VPN tab. *Even if they match and both are set to SHA, you might try changing to MD5 if you can't find anything else wrong -- some peers have a flaky SHA implementation.* * Possible mismatch in encryption domains - do both sides match in terms of subnets? If your partner is a PIX or another Checkpoint (or any reasonably strict product) phase 1 will fail if the two sides mismatch in terms of local/remote networks   If, for example, you have your local network precisely defined as "1.2.3.0/29" and and your peer has it loosely defined as "1.2.3.0/24" they mismatch and the tunnel will fail.  If, for example, you have your local domain defined as a network of "1.2.3.0/29" and and your peer has it defined as individual hosts within that network, they mismatch and the tunnel will fail.  The same is true for the definitions of the remote network.  Your local nets must match the peers remote nets  Your remote nets must match the peer's local nets.   * DH Group mismatches: Especially if your partner is a PIX, try having PIX use group 1 vs. group 2. You can't specify whether your 4.1 machine will use group 1 or group2. It autodetects. It often autodetects wrong, and believes group 2 traffic to be group 1.   If your partner is a Nortel, and the previous suggestions didn't help, you might try:   * to enable BOTH MD5 and SAH1 on your side * to use type/group 2 vs type/group 1 (or 7) on the Nortel side * to mark the tunnel as "nailed up" on the Nortel |
|  | VPN failure with loop through:   * Seemingly good Phase 1 completion * Seemingly good Phase 2 completion * Delete SA received from peer   over and over again | This is very frustrating. It hasn't happened to me often. It 's obviousIy making it through phase 1, so you'd expect the answer to lie in phase 2.  Nevertheless, the times I've seen it, it's always ended up being a phase 1 problem (except with a Nortel, where it's really been some sort of issue on their side)  Problems I've seen cause this, In order of likelihood:   * One or both firewalls support subnets, and the subnet masks or networks mismatch * Partial overlap in the encryption domains you have defined for your peer gateways * If partner is a Nortel try marking tunnel nailed up on the Nortel. This is a kludge, but it may fix the problem for the moment. |
|  | Checkpoint log message of:  Encryption failure. Peer used wrong methods: Scheme IKE | Mismatch in encryption algorithm, hash method or PFS on rulebase (not either peer object) encryption properties |
|  | Checkpoint log message of:  No common authentication methods between myself and peer | Pre-shared key mismatch |
|  | Securemote failure with Checkpoint log message of  "User not properly defined" | Encryption mismatch (e.g. 3DES vs DES) |
|  | VPN failure with:   * Good Phase 1 completion * Good Phase 2 completion * No response from peer | The tunnel itself is probably fine, with the traffic being blocked by firewall rules on peer's network |

You're using a Checkpoint NG Box

Many 4.1 symptoms are identical for NG. Look at 4.1 symptoms as well

|  |  |  |
| --- | --- | --- |
| Platform | Symptom/Message | Likely cause or solution |
| Checkpoint NG | Checkpoint log message of  "Tunnel failure, cannot find IPSec methods of the community (VPN Error code 01)"  even though you're using tradtional mode and there are no communities | A rule with encrypt action defined specifies a peer which is the same as the gateway on which the rule is installed.  In other words, you've mistakenly specified yourself (or some other box included in the install scope) as the remote gateway. When I did this, it was because I accidentally selected the wrong "allowed peer" from the drop-down list, and I felt really dumb. |
|  | Checkpoint log message of  "Cannot identify peer for encrypted connection; (VPN Error code 01)" | The times (once or twice) that we've seen this, it seems to mean "I have this peer defined with correct phase 1 parameters. Perhaps SOME subnets or hosts can be negotiated correctly with this peer, but THIS particular address is not in the Checkpoint's encryption domain." |
|  | Checkpoint log message of  "Encryption fail reason: Cannot identify peer for encrypted connection; (VPN Error code 02)" | Network address translation to an IP address not included in the VPN domain is being applied to the source IP address of the site to site VPN access attempt. i.e. you are NAT'ing your source address to something that isn't defined in your local encryption domain. |
|  | Checkpoint log message of  encryption failure: decrypted methods didn't match rule (VPN Error code 03) | Probably, you are specifying the wrong encrypton, authentication, or PFS on the encrypt action in your rulebase  Much less likely, you've made a deeper error, and have overlapping encryption domains. You can check for this on the EMC by issuing the "vpn overlap\_encdom" command |
|  | Checkpoint log message of  "Packet is dropped because there is no valid SA - please refer to solution sk19423 in SecureKnowledge Database for more information". | Obviously, there's no valid SA. It's an Unhelpful Message  Looking at sk149423 is a waste of time. It's also unhelpful. But you'll look at it anyway.  Not much guidance I can give you here except to note that this must mean one of two things:  Either an outgoing packet needs to be encrypted but a new IPSec SA needed for its encryption could not be created; or an incoming packet needs to be decrypted but the IPSec SA matching the SPI on the packet does not exist  This is most likely some kind of out-of-sync condition where one peer is stubbornly holding on to the notion that it has a valid SA for the tunnel and the other peer insists that there is no SA. Your best bet is to somehow forcibly clear the SA's on both sides.  Reinstalling the policy should clear the SA's on a Checkpoint.  On a PIX, the commands are  clear crypto ipsec sa  clear crypto isakmp sa    Of course, doing so will knock down any OTHER tunnels that are up and force THEM to renegotiate as well  If it's your partner who's holding on the bad (stale) SA, NOTHING you can do will fix it. Either they have to fix it, or it will eventually (hours, maybe days) time itself out. |
|  | Your partner is a Cisco 3000 VPN concentrator.  Your CP log cycles through endless and rapid  Main Mode completion Received message from peer: Delete SA | Stale SA's on the VPN 3000 peer. You can't fix this They have to.  For them to clear these, they need to go to Administration --> Administer sessions, and from this window , select the session with your peer and click "log out" |
|  | Any situation where your partner says they see complaints about "peer id" or messages like 'We require peer to have id “x.x.x.x” but peer declares “y.y.y.y” ' | It's looking for you to send an "ISAKMP identity string" identifying your firewall as a (supposedly optional) part of the negotiation. While 4.1 would ignore the request, NG will send back the IP address the Checkpoint has on its "general" properties tab.  Ideally, have the netscreen not look for one, less ideally, have them try putting in the IP address the Checkpoint has on its "general" properties tab, even if this IP is some private address not reachable via the VPN -- it's just looking to match an ID string.  Do NOT try and solve this by changing the IP on the general tab to match the outside IP! That is likely to cause more problems than it solves: killing your existing VPNs and even making the firewall unmanageable from the GUI. |
|  | Your partner is a Netscreen (or possibly other) peer.  On your Checkpoint you see: "main mode completion" then silence | Problem with Netscreen "peer ID" for your firewall,  It's looking for you to send a string identifying your firewall as a (supposedly optional) part of the negotiation. 4.1 will ignore the request, which the netscreen may or may not be willing to live with. NG will send back the IP address the Checkpoint has on its "general" properties tab.  Ideally, have the netscreen not look for one, less ideally, have them try putting in the IP address the Checkpoint has on its "general" properties tab, even if this IP is some private address not reachable via the VPN -- it's just looking to match an ID string. |
|  | Your partner is a Symantec SGS, possibly others.  You get a Checkpoint log message of  IKE: Phase 1 Received notification from Peer: payload malformed | This is how the SGS responds to a "peer ID" problem.  NG will send back the IP address the Checkpoint has on its "general" properties tab.  Ideally, have the netscreen not look for one, less ideally, have them try putting in the IP address the Checkpoint has on its "general" properties tab, even if this IP is some private address not reachable via the VPN -- it's just looking to match an ID string. |
|  | Your partner is a PIX. In the Checkpoint log, you see:  IKE: Main Mode Completion  reason: Client Encryption: User Unknown  OM: Failed to obtain user object or unknown user  Despite the fact that this is obviously not a SecuRemote client attempting to connect via a client VPN. | This is a misconfiguration on the PIX side. The PIX is using dynamic or client VPNs for some other connection, and is getting confused. You can do nothing, it must be fixed on the PIX. See below in the PIX section for suggestions to give your counterpart. |
|  | Your peer is another NG machine.  You see a VPN failure with the message  "Cannot calculate IKE ranges" | Don't try and NAT the remote addresses on your NG box --i.e. add a "no translation" NAT rule for the network objects in your remote encryption domain going through the tunnel on your side |
|  | Your partner is a Nokia Crypto Cluster. Things look fine on your end. The person configuring the Cluster says they get a message of  "terminated by state machine" | This is the Crypto Cluster's way of complaining about an ISAKMP identity issue. It's looking for you to send a string identifying your firewall as a (supposedly optional) part of the negotiation. See above.  Ideally, configure the Crypto Cluster not to look for one, less ideally, try putting in the IP address the Checkpoint has on its "general" properties tab, even if this IP is some private address not reachable via the VPN -- it's just looking to match an ID string. |
|  | Your partner is a Nokia Crypto Cluster. The partner says they see a "tunnel come up" on their Nokia | They only mean they see at least a phase 1 completion. No promises about phase 2 |

You're using a Nortel

|  |  |  |
| --- | --- | --- |
| Nortel | Nortel log message of:  isakmp[13] invalid id information in message from *x.x.x.x* | This is the same issue about "peer IDs" and "isakmp identities" that one sees on Checkpoints  Is "vendor ID" disabled for the tunnel on the Nortel? "Vendor ID" on the Nortel just means "ISAKMP identity string."  If your partner is a Checkpoint, refer to the NG section above for things they or you can do  If partner is a PIX, make sure that their "isakmp identity address" is explicitly specified on the PIX. |

You're using a Cisco Box

|  |  |  |
| --- | --- | --- |
| Platform | Symptom/Message | Likely cause or solution |
| PIXs and Cisco routers | (Router) log message of:  CRYPTO-4-IKMP\_BAD\_MESSAGE: IKE message from *x.x.x.x* failed its sanity check or is malformed | Pre-shared key mismatch  This is more often a router VPN message than a PIX VPN message |
|  | (Router) log message of  CRYPTO-6-IKMP\_MODE\_FAILURE: Processing of Main Mode failed with peer at *x.x.x.x* | No policy defined with correct combination of DES/3DES, MD5/SHA and Group1/2  This is more often a router VPN message than a PIX VPN message |
|  | Any message containing a reference to "proxy identities not supported" usually something like the following PIX debug output:  IPSEC(validate\_proposal\_request): proposal part #1, (key eng. msg.) dest= 1.1.1.2, src= 2.2.2.3, dest\_proxy= 4.4.4.0/255.255.255.192/0/0 (type=4), src\_proxy= 5.5.5.0/255.255.255.224/0/0 (type=4), protocol= ESP, transform= esp-3des esp-md5-hmac , lifedur= 0s and 0kb, spi= 0x0(0), conn\_id= 0, keysize= 0, flags= 0x4 IPSEC(validate\_transform\_proposal): proxy identities not supported  where 1.1.1.2 and 2.2.2.3 are the VPN peer addresses, and 4.4.4.0/26 and 5.5.5.0/27 are the local and remote networks (encryption domains for each peer)  You MUST run debug on the PIX to see these messages | The most common PIX problem.  The two peers must agree exactly on the definitions for the local and remote networks (i.e the encryption domains for each peer)  If, for example, you have your local network precisely defined as "1.2.3.0/29" and and your peer has it loosely defined as "1.2.3.0/24" they mismatch and the tunnel will fail.  If, for example, you have your local domain defined as a network of "1.2.3.0/29" and and your peer has it defined as individual hosts within that network, they mismatch and the tunnel will fail.  The same is true for the definitions of the remote network.  Your local nets must match the peers remote nets  Your remote nets must match the peer's local nets.  Check the dest\_proxy and src\_proxy reported in the debug message. Compare them against the network objects specified in your VPN ACL.  I should also note that "proxy identities not supported" can come up if you've specified particular ports on the "interesting traffic" ACL, and the traffic doesn't match the specified ports. I would expect "denied" instead, but no, it's "proxy identities not supported." This, however, is very easy to debug by simply making the ACL "permit ip source dest " and "permit ip dest source.". If that works and your desired ACL doesn't, then the restrictions must be the issue. |
|  | PIX debug output of:  IPSec(validate\_transform\_proposal): proxy identities not supported ISAKMP: IPSec policy invalidated proposal ISAKMP (0:2): SA not acceptable! | Usually pix-to-pix, but can happen with other firewalls smart enough to do detailed negotiation, like a checkpoint. The official wording from Cisco is that "The access list on each peer should mirror each other (all entries should be reversible)."  The key here is that this is just more info associated with the "proxy identities" message described above. |
|  | PIX debug output of:  ISAKMP (0:1); no offers accepted! ISAKMP (0:1): SA not acceptable! | No policy on PIX with correct combination of DES/3DES, MD5/SHA and Group1/2 |
|  | PIX debug output of:  IPSEC(validate\_proposal): invalid local address *x.x.x.x* ISAKMP (0:3): atts not acceptable. Next payload is 0 ISAKMP (0:3): SA not acceptable! | Cisco says that "The crypto map *map-name local-address interface-id* command causes the router to use an incorrect address as the identity because it forces the router to use a specified address."  Which means that either the crypto map is applied to the wrong interface or, it is not applied at all. Check the configuration to ensure that crypto map is applied to the correct interface. |
|  | PIX debug output of:  IPSec (validate\_proposal): transform proposal(port 3, trans 2, *hmac\_alg 2*) not supported ISAKMP (0:2) : atts not acceptable. Next payload is 0 ISAKMP (0:2) SA not acceptable | Mismatch in the PIX "crypto ipsec transform-set" statement for this tunnel |
|  | PIX debug output of:  ISAKMP: No cert, and no keys (public or pre-shared) with remote peer *x.x.x.x* | Wrong peer address in PIX "crypto map" statement for this tunnel |
|  | PIX debug output of:  IPSEC(validate\_transform\_proposal): peer address *x.x.x.x* not found | Cisco's only doc on this message says:  *The error message below normally appears with the corresponding VPN 3000 Concentrator error message "Message: No proposal chosen(14)". This is a result of the connections being host-to-host. The router configuration had the IPSec proposals in an order such that the proposal chosen for the router matched the access list, but not the peer. The access list had a larger network that included the host that was intersecting traffic. To correct this, make the router proposal for this concentrator-to-router connection first in line, so that it matches the specific host first.*  This makes little sense to me in terms of a PIX, and attempting to interpret this explanation for a PIX has never helped me.  From experience, though, If *x.x.x.x* is the address of your own firewall, check and see if you haven't accidentally reversed an ACL. Look at the way that they are mirrored (vs identical) in the Cisco PIX Firewall and VPN Configuration Guide Chapter 7 |
|  | PIX debug output of:  IPSEC(initialize\_sas): invalid proxy IDs | The received proxy identity does not match the configured proxy identity as per the access list, i.e. the gateways are right but the host isn't in the ACL. You've established a tunnel, and then the peer tries to send you some traffic through it that doesn't match the "interesting traffic"/"encryption domains" specified on your side. |
|  | PIX debug output of:  Reserved Not Zero on Payload 5 | Almost always an ISAKMP key mismatch  Can also show up if you've accidentally cut and pasted the wrong peer address into the crypto map |
|  | PIX debug output of:  Reserved Not Zero on Payload 8 | Often, a mismatch in encryption domains Make sure your definitions for networks/hosts on both sides match the peer's  By report, this can sometimes indicate that you need to add "no-xauth no-config-mode" to the isakmp key line.  This is not necessarily a fatal error - sometimes it's a stupid peer that won't follow protocol. I have working tunnels that get this message all the time |
|  | PIX debug output of:  Reserved Not Zero on Payload 11 | The one time I've seen it, it was an ISAKMP key mismatch  Can also show up if you've accidentally cut and pasted the wrong peer address into the crypto map |
|  | PIX debug output of:  ISAKMP: error, msg not encrypted | The one time I've seen it, it was an ISAKMP key mismatch with a concentrator |
|  | PIX debug output of:  VPN Peer:ISAKMP: Peer Info for *x.x.x.x*/500 not found - peers:0 | No tunnel has ever been set up, but some operation is being attempted that presumes one has been set up, Maybe you got a delete SA or delayed traffic from a misconfigured peer? If this shows up alongside "retransmitting phase 1" see below. |
|  | PIX debug output of:  ISAKMP (0): retransmitting phase 1. | If you are initiating, you sent a phase one and got no response. Do both peers have a correct route out? Is one one the other getting its IKE traffic blocked by some intervening firewall or ACL'ed router?  If s/he is initiating, Peer started a phase 1 and you answered, but it never completed. Your PIX is still trying. Doesn't tell you much, but in the absence of other errors, it indicates your side is not as likely to be the problem |
|  | PIX debug output of:  return status is IKMP\_NO\_ERROR crypto\_isakmp\_process\_block:src:*x.x.x.x*, dest:*y.y.y.y* spt:500 dpt:500  VPN Peer:ISAKMP: Peer Info for *x.x.x.x*/500 not found - peers:z  ISAKMP (0): retransmitting phase 1... crypto\_isakmp\_process\_block:src:*x.x.x.x*, dest:y.y.y.y spt:500 dpt:500 VPN Peer:ISAKMP: Peer Info for *x.x.x.x*/500 not found - peers:z  x.x.x.x is his peer y.y.y.y is yours | This has appeared where the partner is a Cisco device and there was a mistake in the partner's NAT translation for the address of the firewall (or destination hosts) on our side |
|  | PIX debug output of:  ISADB: reaper checking SA *foo*conn\_id = 0 | Ignore it. Normal message. You'll see lots of them. This is just garbage collection looking for stale SA's to clean up |
|  | PIX debug output of:  ISAKMP (0): processing NOTIFY payload 26 protocol 1 spi 0, message ID = *foo* return status is IKMP\_NO\_ERR\_NO\_TRANS ISAKMP (0): retransmitting phase 2... | Well, phase one has completed, but phase 2 is failing. Most commonly, this is just another manifestation of mismatched encryption domains, where you have a network specified and s/he has a single host |
|  | PIX debug output of:  ERROR: unable to fragment packet pktsize=*foo*, eff\_mtu = *bar* | Probably not the source of your problem -- this is usually just a sign that something is negotiating the maximum MTU |
|  | PIX debug output of:  ISAKMP (0): processing SA payload. message ID = <something> ISAKMP : Checking IPSec proposal 1 ISAKMP: transform 1, <something> ISAKMP: attributes in transform: ISAKMP: SA life type in seconds ISAKMP: SA life duration (basic) of 3600 ISAKMP: encaps is 1 ISAKMP: authenticator is HMAC-MD5IPSEC(validate\_proposal): transform proposal (prot 3, trans 2, hmac\_alg 1) not supported  ISAKMP (0): atts not acceptable. Next payload is 0 | Mismatch between your transform-set and peer's, or your transform-set is somehow invalid |
|  | Normal-looking IPSEC(initialize\_sas): , messages  no IKMP\_NO\_ERR message  then  IPSEC(sa\_initiate): ACL = deny; no sa created IPSEC(sa\_initiate): ACL = deny; no sa created IPSEC(sa\_initiate): ACL = deny; no sa created <etc> | Probably, the peer is specifying single hosts on his key exchange, while you have a subnet specified. If peer won't turn on key exchange for subnets (Checkpoint) or expand ACL's to match your subnet (PIX) then you'll have to change your ACL's to contain each lousy host.  Can also happen if you've simply specified one or he other encryption domain incorrectly (typo?) or specified the wrong "match " or peer in the crypto map (bad cut'n'paste)  Note that this is an IPSec message. Very possibly, there's already a good ISAKMP SA, and you will not see any additional ISAKMP traffic during debug -- just the annoying repeated message. Possibly there's an "incomplete" ISAKMP SA in memory that you won't even see with a "sho crypto isakmp sa" command.  On 6.3(1) I have once seen this message in a PIX-to-PIX setup, when the PIX believed there was a mismatch because I had specified  object-group network foo network-object host 1.2.3.4  instead of  object-group network foo network-object 1.2.3.4 255.255.255.255  which looks like a bug to me. |
|  | No phase one messages seen at all  Nothing but  IPSEC(sa\_initiate): ACL = deny; no sa created IPSEC(sa\_initiate): ACL = deny; no sa created IPSEC(sa\_initiate): ACL = deny; no sa created  and no other messages | I really think this is the same thing as when one see normal phase one messages, and then sees this afterwards.  This drives me nuts. I'm going to have to get a sniffer out and prove what's going on. In this case, you never see ANY kind of ISAKMP messages, or any other IPSec messages. That looks as though you are never sending anything out -- that it's trappped on your side and you never even try and negotiate a tunnel  But I'm damned if I don't think that the traffic is going on and the messages just aren't displaying. Because every time I've seen it, it's always been a subnet mismatch that caused it. If you control both ends then it's fairly easy to compare the VPN ACL's with a "sho access list *foo*" on both sides and go through them line by line.  If you control only one endpoint and a have a recalcitrant person running the other, who insists their side is completely correct, then good luck.  All I can do is to repeat that every single time I have ever seen this, a subnet mismatch was the cause, even though there were no ISAKMP or IPSec messages to indicate the two machines were in communication.  Interestingly enough, this "no other messages" condition has happened to me only when I had IOS boxes on both ends, which makes me think that the two must have some comm going that just doesn't show in the debug.  Note: I had this happen to me this afternoon, and the root cause was me trying to be tricky. I had a subnet 10.0.0.0/28 call it, that had been expanded to 10.0.0.0/27. I had an existing network object like so:  object-group network foo network-object 10.0.0.0 255.255.255.240  I was lazy and tried to make the same object work in both tunnels using the old definition and the new one, so I made it:  object-group network foo network-object 10.0.0.0 255.255.255.240 network-object 10.0.0.0 255.255.255.224  I've had that kind of trick work many times with things like:  object-group network foo network-object 10.0.0.0 255.255.255.240 network-object host 10.0.0.1  when the other guy insisted on using hosts. Well, today, it broke the tunnel. Fine, I was cheating anyway, but the point is that even in the absence of other debug messages, the two had to be talking for either side to know there was a mismatch.  This morning's incarnation of this bug is nothing which isn't implied by the foregoing, but is worth a note. The Checkpoint peer included its own external IP address in its encryption domain. We didn't want to acess it, and in fact rules on our inside interface disallow any such traffic. Nevertheless, the tunnel failed with this error up until we added that gateway address into our remote netwoks object for the VPN ACL. They have to match *even if* the traffic will never flow.  An unconfirmed report from the mailbag tells of a tunnel problem between a PIX 515 and a Cisco 1841. In this case, even having the maps identically defined with  network-object 172.20.0.0 255.254.0.0  didn't work. It seems that the 1841 was internally splitting the "172.20.0.0/255.254.0.0" into individual class C's (Class-based setup, maybe?) and the VPN failed until the pix side was defined as  network-object 172.20.0.0 255.255.0.0 network-object 172.21.0.0 255.255.0.0  And then there's this:  access-list random\_vpn permit ip object-group my\_local object-group his-remote access-list random\_vpn permit ip object-group his-remote object-group my\_local  generated the acursed "IPSEC(sa\_initiate): ACL = deny; no sa created". But it got fixed by reversing it to:  access-list random\_vpn permit ip object-group his-remote object-group my\_local access-list random\_vpn permit ip object-group my\_local object-group his-remote |
|  | PIX debug output of:  ISAKMP (0): processing NOTIFY payload 36136 protocol 1 spi 0, message ID = 15262787 ISAMKP (0): received DPD\_R\_U\_THERE from peer *x.x.x.x* ISAKMP (0): sending NOTIFY message 36137 protocol 1 return status is IKMP\_NO\_ERR\_NO\_TRANS | Normal message, not a sign of a problem. Your peer has set a "keepalive" (i.e. nailed the tunnel up). This message just means "peer x.x.x.x checked to see if I was still alive." |
|  | PIX debug output of:  return status is IKMP\_NO\_ERROR ISAKMP (0): sending phase 1 RESPONDER\_LIFETIME notify ISAKMP (0): sending NOTIFY message 24576 protocol 1 VPN Peer: ISAKMP: Added new peer: ip:x.x.x.x/500 Total VPN Peers:8 VPN Peer: ISAKMP: Peer ip:x.x.x.x/500 Ref cnt incremented to:1 Total VPN Peers:8 crypto\_isakmp\_process\_block:src:x.x.x.x, dest:y.y.y.y spt:500 dpt:500 ISAKMP (0): processing DELETE payload. message ID = 2096747792, spi size = 16 ISAKMP (0): deleting SA: src x.x.x.x, dst y.y.y.y return status is IKMP\_NO\_ERR\_NO\_TRANS  ISADB: reaper checking SA 0x11ac374, conn\_id = 0 DELETE IT!  VPN Peer: ISAKMP: Peer ip:x.x.x.x/500 Ref cnt decremented to:0 Total VPN Peers:8 VPN Peer: ISAKMP: Deleted peer: ip:x.x.x.x/500 Total VPN peers:7 | Hmm, could be anything on the peer side. Be sure to explicitly specify "isakmp identity address" before doing much more. |
|  | Partner is a VPN concentrator.. PIX debug messages are :  ISAKMP (0): beginning Main Mode exchange crypto\_isakmp\_process\_block:src:9.1.2.3, dest:10.4.5.6 spt:500 dpt:500 OAK\_MM exchange ISAKMP (0): processing SA payload. message ID = 0 <snip> crypto\_isakmp\_process\_block:src:9.1.2.3, dest:10.4.5.6 spt:500 dpt:500 OAK\_MM exchange <snip> ISAKMP (0): speaking to a VPN3000 concentrator <snip> return status is IKMP\_NO\_ERROR  but no connect | You've completed phase 1 OK, the eror is somewhere in phase 2 |
|  | Phase one goes OK, you see NOTHING for phase two. | Difficult to debug, of course. Silence always is. MOST likely, your partner has things fouled up. But let me note some weird things that I've seen cause this:  A dual-homed Windows Server 2003 partner caused this when he routed traffic to my VPN peer out of the correct interface, and traffic for the actual host out the wrong interface.  Another dual-homed Windows Server 2003 partner caused this when he had two default gateways specified (thinking he needed one for each interface) and created essentially the same situation.  I once caused this on the PIX side by accidenatlly specifying a network IP as a host in my objects, i.e.  object-group network partner\_net network-object host 10.1.1.0  when I meant  object-group network partner\_net network-object 10.1.1.0 255.255.255.0 |
|  | PIX debug output of:  ISAKMP (0): processing DELETE payload. message ID = 1166168095, spi size = 4 IPSEC(key\_engine): got a queue event... IPSEC(key\_engine\_delete\_sas): rec'd delete notify from ISAKMP  VPN Peer: IPSEC: Peer ip:x.x.x.x/500 Decrementing Ref cnt to:2 Total VPN Peers:1 VPN Peer: IPSEC: Peer ip:x.x.x.x/500 Decrementing Ref cnt to:1 Total VPN Peers:1 return status is IKMP\_NO\_ERR\_NO\_TRANS | Not an error, per se. Your peer just sent you a "delete ipsec sa" instruction |
|  | PIX debug output of:  crypto\_isakmp\_process\_block:src:x.x.x.x, dest:32.96.134.83 spt:500 dpt:500 ISAKMP (0): processing DELETE payload. message ID = 3415178296, spi size = 16 ISAKMP (0): deleting SA: src x.x.x.x, dst y.y.y.y return status is IKMP\_NO\_ERR\_NO\_TRANS ISADB: reaper checking SA 0xf8048c, conn\_id = 0 DELETE IT! | Not an error, per se. Your peer just sent you a "delete isakmp sa" instruction |
|  | All VPN messages look good. The PIX logs show a "NO TRANS" error | This is a NAT issue, not a VPN issue. See my preliminary notes towards a page on that [here.](http://www.rick-thompson.com/misc/pix-outgoing.htm) |
|  | All VPN messages look good. The PIX logs show a translation being built. The connection dies with a SYN timeout | If you are sure that the VPN is all good, then this is rourting or firewalling somewhere beyond your own VPN gateway. |
|  | Your partner is a Checkpoint. In his/her logs, your counterpart sees  IKE: Main Mode Completion  reason: Client Encryption: User Unknown  OM: Failed to obtain user object or unknown user  Despite the fact that this is obviously not a SecuRemote client attempting to connect via a client VPN. | This is a misconfiguration on the PIX side. The PIX is using dynamic or client VPNs for some other connection, and is getting confused.  If your crypto map contains both static and dynamic entries, alway give ALL of the statics a LOWER sequence number than ANY of the dynamics.  If any of your isakmp keys are wildcarded it should see the non-wildcard entries FIRST  Add "no-xauth no-config-mode" to the isakmp key statement for the gateway-to-gateway peer |
|  | Your partner is a Nokia Crypto Cluster. Things look fine on your end. The person configuring the Cluster says they get a message of  "terminated by state machine" | This is the Crypto Cluster's way of complaining about an ISAKMP identity issue. It's looking for you to send a string identifying your firewall as a (supposedly optional) part of the negotiation. See above.  Look for "message 24576" debug on the PIX.  If found, make sure that "isakmp identity address" is explicitly specified on the PIX.  The PIX will send back either its hostname, or the IP address of the isakmp interface depending on your config line for "isakmp identity" |
|  | Your partner is a Nokia Crypto Cluster. The partner says they see a "tunnel come up" on their Nokia | They only mean they see at least a phase 1 completion. No promises about phase 2 |
|  | Tunnel comes up, initial contacts are OK, client fails on large packets | Someone, somewhere has not accounted for the overhead added by the VPN. I.e., the packet size plus the bytes added for the VPN encapsulation give you packets too big for ethernet, but which are marked "don't fragment"  You can throttle this back on your side with:  sysopt connection tcpmss 1350 |

You're using an off-brand box

|  |  |  |
| --- | --- | --- |
| Platform | Symptom/Message | Likely cause or solution |
| Raptor | Raptor messages to the effect of  failed to locate an isakmp sa | This just means that phase one has either failed, or that the Raptor has deleted the SA |
| Sidewinder | All looks well on the non-sidewinder side, and phase 1 seems to complete OK, but no connection | Sidewinders will reject the phase 2 proposal if the key lifetimes are mismatched. The default key lifetime for a sidewinder is 700 seconds |
| Any | Symptom: Partner's firewall is running Windows. You see lots of netbios-ns traffic hitting you from his gateway, but no IKE handshaking ever completes | Timeouts while windows tries to perform NBT name resolutions -- have them add the address of your gateway with some name, any name to their hosts or lmhosts file |
|  | Symptom: Your partner is running a Raptor. You see no traffic at all | Raptors are extremely sensitive to giving up or keeping bad SA's. Basically the Raptors will need to "reset" their tunnels before each attempt |

Some Handy PIX / IOS syntax reminders

Cisco show comands:

show crypto isakmp sa

This command shows the ISAKMP SA built between peers

show crypto ipsec sa

This command shows IPSec SA built between peers

show crypto engine connection active

(IOS only) This command shows each Phase 2 SA built and the amount of traffic sent. Remember that Phase 2 SAs are uni-directional, so each SA will show traffic in one direction only (encryptions are outbound, decryptions are inbound).

Coaxing debug output out of a PIX:

By the book, PIX's won't output crypto debug info to a telnet/ssh session. It's possible to get them to, and here's how:

Open a sesson to the PIX. Enable, and issue

debug crypto isakmp  
debug crypto ipsec  
debug crypt engine

do a "write mem" (I don't know for sure that this is required, but it sometimes seems to be)

Go into config t and issue:

term mon

OK, fine, no debug output. BUT then go and open a SECOND session. Do a "term mon" there as well,

In trying to figure out how to handle the debug stream, the PIX forgets that it isn't supposed to send crypto debug to a telnet/ssh session, and will begin sending the crypto debug to the FIRST session. If you don't see debug, log out of sesson 1 altogether and start a third one in its place

WARNING: This is taking advantage of a bug. It may stop working on some new release.

WARNING: Once you have this going, it will output to a new session on connection -- *before* authentication if it's a telnet session. Another reason to use ssh and not telnet, since using ssh will require the authentication BEFORE it starts sending debug info to the session's virtual console.

Clearing your existing SA's:

PIX:

clear crypto ipsec sa

clear crypto ipsec sa peer x.x.x.x

clear crypto ipsec sa map foo

clear crypto isakmp sa

Checkpoint:

Reinstall the policy

Misc

Packet debug syntax:

debug packet *if\_name* [src *x.x.x.x* [netmask *a.a.a.a]*] [dst *y.y.y.y* [netmask *b.b.b.b*]] [[proto icmp]|[proto tcp|udp [sport *xxx*][dport *xxx*]][rx|tx|both]

A sample PIX config:

Just the lines relating to the VPN

Peer at location 1

hostname location-1-peer

nameif ethernet0 outside security0

nameif ethernet1 inside security100

object-group network location-1

network-object 10.0.1.0 255.255.255.0

object-group network location-2

network-object 10.0.2.0 255.255.255.0

access-list acl\_outside deny ip any any

access-list acl\_inside permit ip object-group location-1 object-group location-2

access-list acl\_inside deny ip any any

access-list acl\_vpns\_100 permit ip object-group location-1 object-group location-2

ip address outside 192.168.1.1 255.255.255.0

ip address inside 10.0.1.1 255.255.255.0

nat (inside) 0 0.0.0.0 0.0.0.0 0 0

static (inside,outside) 10.0.1.0 10.0.1.0 netmask 255.255.255.0

access-group acl\_outside in interface outside

access-group acl\_inside in interface inside

sysopt connection tcpmss 1350

sysopt connection permit-ipsec

crypto ipsec transform-set 3des-sha1 esp-3des esp-sha-hmac

crypto map vpns 100 ipsec-isakmp

crypto map vpns 100 match address acl\_vpns\_100

crypto map vpns 100 set peer 192.168.2.1

crypto map vpns 100 set transform-set 3des-sha1

crypto map vpns interface outside

isakmp enable outside

isakmp key s0m3shar3dk3y address 192.168.2.1 netmask 255.255.255.255

isakmp identity address

isakmp policy 10 authentication pre-share

isakmp policy 10 encryption 3des

isakmp policy 10 hash sha

isakmp policy 10 group 2

isakmp policy 10 lifetime 86400

Peer at location2

hostname location-2-peer

nameif ethernet0 outside security0

nameif ethernet1 inside security100

object-group network location-1

network-object 10.0.1.0 255.255.255.0

object-group network location-2

network-object 10.0.2.0 255.255.255.0

access-list acl\_outside deny ip any any

access-list acl\_inside permit ip object-group location-2 object-group location-1

access-list acl\_inside deny ip any any

access-list acl\_vpns\_100 permit ip object-group location-2 object-group location-1

ip address outside 192.168.2.1 255.255.255.0

ip address inside 10.0.2.1 255.255.255.0

nat (inside) 0 0.0.0.0 0.0.0.0 0 0

static (inside,outside) 10.0.2.0 10.0.2.0 netmask 255.255.255.0

access-group acl\_outside in interface outside

access-group acl\_inside in interface inside

sysopt connection tcpmss 1350

sysopt connection permit-ipsec

crypto ipsec transform-set 3des-sha1 esp-3des esp-sha-hmac

crypto map vpns 100 ipsec-isakmp

crypto map vpns 100 match address acl\_vpns\_100

crypto map vpns 100 set peer 192.168.1.1

crypto map vpns 100 set transform-set 3des-sha1

crypto map vpns interface outside

isakmp enable outside

isakmp key s0m3shar3dk3y address 192.168.1.1 netmask 255.255.255.255

isakmp identity address

isakmp policy 10 authentication pre-share

isakmp policy 10 encryption 3des

isakmp policy 10 hash sha

isakmp policy 10 group 2

isakmp policy 10 lifetime 86400

Common mistakes in no particular order

1. When defining PIX crypto maps, you can have as many "named" maps as you want, but only one named map can be applied to a given interface at any one time. To do multiple tunnels over the same interface, you use a single named map, and then different sequence numbers within that map. The map is searched in sequence order for a match.
2. When a single PIX named crypto map contains both static and dynamic entries, alway give ALL of the statics a LOWER sequence number than ANY of the dynamics.

Out into the weeds

*Things I think are true, but can't swear to*

PIX VPN Interesting traffic vs. interface ACL's

Version 7 may work a bit differently, but I'm still playing there.

For discussion, assume a PIX with two interfaces, inside, and outside: inside being some secure network, and outside being some non-secure network across which one wishes to communicate via VPN. IKE/IPSec control statements are applied as follows:

sysopt connection permit-ipsec  
crypto map foo interface outside  
isakmp enable outside

Cisco's note in the PIX 6.3 Command Reference (under the "crypto map" command) that "The crypto access-list is not used to determine whether to permit or deny traffic through the interface. An access list applied directly to the interface with the access-group command makes that determination." notwithstanding, experimentation shows me that what actually happens is:

1. The 3 IKE/IPSec control statements above create an implied "allow any IKE/IPSec traffic" rule on the outside interface.  
   This "implied rule" is matched first by any encrypted packet incoming on the outside interface.
2. If it is matched, no further ACL processing takes place for the packet. Traffic matching this implied rule then bypasses any other ACL on the interface and is evaluated against the "interesting traffic" ACL.
3. If the does not match the interesting traffic list, and the correct peer, it's dropped with a "proxy identities" message.
4. Note that this means that ACLs applied inbound to the outside interface are irrelevant to the VPN traffic.

if one applies ACLs as follows:

access-list deny\_all deny ip any any  
access-group deny\_all in interface outside

Properly encrypted traffic matching the interesting traffic ACL (and from the correct peer) will still be allowed inbound on the outside interface. This is currently my config on [deleted]

Why does this work? It works, if and only if, you have "sysopt connection permit-ipsec" specified.

If you use that sysopt, "allows" and denies in an ACL applied to the outside interface are IGNORED for VPN traffic.

Cisco's note should, I think, have said ""The crypto access-list is not used to determine whether to permit or deny NON-VPN traffic through the interface. An access list applied directly to the interface with the access-group command makes that determination. An access list applied directly to the interface will govern VPN traffic if sysopt connection permit-ipsec is NOT set. If sysopt connection permit ipsec IS set, then an ACL applied to the outside interface is IGNORED for VPN traffic.

ACLs applied to the inside interface work as expected. I.e. outgoing traffic which arrives inbound on the inside interface must pass any ACL applied inbound. Traffic going outbound to the secure net from the inside interface must pass any ACL applied outbound to the inside interface (though, of course, [we] don't usually use these).

I have not checked the effect of ACLs applied outbound to the outside interface. My suspicion is that these would be treated just like the incoming are for encrypted traffic.

The net is that you cannot limit traffic across the VPN to particular ports by setting "allow all IP" in the interesting traffic list and then placing specific "allows" in an ACL applied to the outside interface iuf you're using permit ipsec.

If you want to limit the traffic in the VPN to specific hosts and ports, it must be either be done in the interesting traffic ACL or you can't have sysopt connection permit ipsec. Note the two peers must still be symmetric. i.e. each peer must be the mirror of the other.

I think this is the intended behavior. See the sample VPN config in the Cisco PIX Firewall and VPN Configuration Guide Chapter 7. Note they have no permits on the outside interface. This by default should deny traffic If things didn't work the way I describe above, their own sample config shouldn't work.

Send comments/additions to [rick@rick-thompson.com](mailto:rick@rick-thompson.com)

Last updated by: rickt on 2005/08/15