



# Understanding PIX Behavior

Peter J. Welcher

## Introduction

In this article we'll look at a couple of the PIX areas which seem to be undocumented, counter-intuitive, or just different. At least, they seem that way to me. For PIX aficionados, bear in mind that I "grew up" on Cisco IOS, which probably explains some of my expectations as to behavior. My justification is that most people came to PIX by way of doing routers first, so that may be a common problem. The article also contains some tips on things I've found useful when working with the PIX.

Areas we'll look at in this article:

- 1 [PIX order of operations](#)
- 1 [PIX Best Practices](#)
- 1 [Split tunneling](#)
- 1 [PIX ACLs and security levels](#)
- 1 [Using NAT internally](#)
- 1 [Controlling inbound VPN user traffic](#)
- 1 [Troubleshooting connectivity through a PIX](#)

We will conclude with a [comment on a reader question](#), one I've had before, on the subject of enabling jumbo frames. Your emailed thoughts on this subject would be welcome!

## PIX Order of Operations

For quite a while, I have been referring people to the Cisco NAT Order of Operations document. For example, see <http://www.netcraftsmen.net/welcher/papers/ipsec2.html>. The Cisco URL is [http://www.cisco.com/en/US/partner/tech/tk648/tk361/technologies\\_tech\\_note09186a0080133ddd.shtml](http://www.cisco.com/en/US/partner/tech/tk648/tk361/technologies_tech_note09186a0080133ddd.shtml).

So when working with a PIX recently, it dawned on me that I had never seen anything like this for the PIX. Is the sequence of operations for an outbound packet like that for Cisco routers: NAT, then ACL, then IPsec encryption?

After some lab testing (using ASDM to speed things up), I have concluded that the answer to that question is "**NO**", the PIX order is **not** the same as in Cisco IOS. The order on an outbound interface, in PIX 7.0, is ACL, then NAT, then IPsec.

I confirmed this by applying an outbound ACL of the form permit ip <test source> any. Here <test source> is the one active interface of the router I was using as a handy packet source. I had already ensured the only route to test destinations from that router was via the PIX, using a static default route. When I use the inside / local address of the source, that is, the actual IP, in the ACL then NAT works, I can ping an outside device and get a reply back. When I use the outside / global address for the source, the one that NAT changes the source IP into, then no NAT translate results -- the ACL blocks the packet before it can be NATted. The traffic is going from inside to outside interfaces using default security levels, so nothing else is blocking it.

I then tested IPsec in relation to NAT. I got an IPsec tunnel up and working from the outside PIX interface to a router interface, with crypto map ACL matching precisely those two addresses. I verified I could ping through the IPsec tunnel. I double-checked encrypted packet counters to make sure the IPsec VPN was being used. When I do Port Address Translation (PAT) to the PIX outside interface, traffic goes via the tunnel. When I do NAT to a different address in the same subnet, ping still works, but the crypto counters no longer increment. (In my lab, I allow non-NAT traffic in and out

the interface.) The conclusion: NAT occurs before the IPsec ACL is consulted, or the IPsec tunnel would not have been used.

## PIX Best Practices

It occurs to me that this latter piece of information may not be as useful as it appears. The good news is, there's a Best Practice lurking there!

I generally prefer to code things so that all my IPsec traffic is excluded from NAT. Thus I do a `"nat (inside) 0 <access-list-name>"` command, where the access-list is the crypto map ACL. That way, traffic receives either NAT or IPsec processing, but never both. It keeps things simpler.

By the way, if you're using ASDM to set up the IPsec, pay attention to the little checkbox at the bottom, the one that says "Exempt PIX side host/network from address translation". ASDM creates a NAT exemption rule entry if you leave the box checked. This is convenient, but if you don't notice what's getting sent to the PIX, the NAT rule may surprise you later when you review the configuration.

## Split Tunneling

I define split tunneling as when you're running a VPN client on your PC, and the IPsec rules allow some unencrypted traffic to go directly to the Internet. I've always thought of it in terms of which traffic bypasses the IPsec VPN tunnel. However, the router and PIX engineers came at this subject from a slightly different perspective, namely, which traffic is protected by the IPsec VPN encryption.

Why does it matter? Well, if you're allowing split tunneling, the router or PIX controls what the PC is allowed to do, when you use the Cisco VPN client. The way you control the split tunneling is in the `isakmp vpn client` configuration block. The access list is written in terms of what traffic the router encrypts to the client, even though (to me) it controls what traffic **in the other direction** is to be encrypted.

With PIX 7.0, the split tunnel access list is a standard access list which specifies sources/destinations to encrypt traffic to. To confirm that this was traffic from the viewpoint of the PC (something the documentation does not clearly state), I tried adding and removing certain subnets. When they were in the list, the PC could not ping them (I presume because the outgoing ping was being sent encrypted and the return traffic was direct, not encrypted. Also noted: if you do `"route print"` on your PC, you can see the networks in the split tunnel list appearing as routes on your PC, when the VPN tunnel is connected.

## PIX ACLs and Security Levels

From a security perspective, you do need to remember that the PIX allows traffic from a more secure to a less secure interface, and replies. If you don't like that, you can always apply an ACL permitting only the outbound traffic you wish.

You do need to specifically allow traffic that you want flowing from a lower security level interface to a higher one. Suppose you apply this access list inbound on the lower security level interface. Bear in mind in doing this that you will then also need to allow outbound traffic to other destinations.

Lesson learned in testing with ASDM: when you disable an ACL rule, it is still present. Thus the ACL is still defined. If you disable the only rule in the ACL, you have the default, which is deny all. Whereas if you remove the ACL from an interface, then you permit whatever is permitted by default. In particular, disabling all rules in an ACL is **not** the same as removing the ACL. Maybe that's obvious, maybe not. I managed to walk into it with the ASDM GUI before having a "doh" moment and catching on to what was happening.

## Using NAT Internally

Apropos of Best Practices, I've seen people doing NAT rules between every pair of interfaces. That's one way to bypass having to create access lists. My recommendation: don't do this. If you do, not only does it make things very hard to understand, but I guarantee it'll get worse over time. Just use a good addressing scheme, no duplicated subnets, and resist the urge to NAT internally. Reserve NAT for the outside world.

## Controlling Inbound VPN User Traffic

Why would one want to control inbound IPsec VPN user traffic? I have run into this sort of situation when I wanted to configure the PIX for multiple IPsec client groups. For example, network admins, server admins, external support people, and general users. The requirement would be for each group to be controlled, so that say network admins could get to network devices, server admins to server subnets, external support people to SAN or PBX or whatever they supported, and general users to email servers and the like.

There are a couple of ways to control inbound IPsec user traffic.

First, beware the convenient command `sysopt connection permit-ipsec`. Testing (PIX 7.0 code) verifies that it does bypass both inbound ACL on the outside (IPsec) interface and outbound ACL on the dmz (or inside) interface. The point to this command is that it gives a "free pass" to all traffic coming out of an IPsec tunnel. It is easy to miss this command when your brain is in access-list mode.

The key to controlling multiple user groups is to assign different VPN address pools for each group. The users will then be assigned addresses that allow you to differentiate between them for access list (ACL) purposes.

If you are doing the above `sysopt` command, you can still control what users can get to via your crypto access list. Basically, only encrypt traffic from allowed subnets to the appropriate user pool. Block any other traffic to the user pools.

It is arguably cleaner to remove the `sysopt` command, and use an access list inbound on the outside (IPsec) interface. Testing shows that this ACL is consulted after the inbound packets are decrypted. You can then explicitly control which packets are allowed in, and to where.

## Troubleshooting Connectivity Through A PIX

When working with PIXes, I've finally learned to not expect to be able to ping the PIX unless I allow `icmp`. For example, there is the majorly insecure:

```
icmp permit any outside
icmp permit any inside
icmp permit any dmz
```

You may find it useful to allow ping from selected addresses to assist troubleshooting. For example, Internet-facing router to outside interface. Inside management subnet to all PIX interfaces.

I've found it useful when setting up a PIX to build in ACL rules specifically permitting ping traffic from a less secure to more secure interface. You can then use ASDM or CLI to disable the rules unless you are troubleshooting. To be more secure, only allow ping from a specified address or range. If you intend to run `traceroute` outbound, you'll need to allow the ICMP replies back in.

If you're doing PAT on the outside interface, `traceroute` outbound still won't work, at least not from a Cisco router. The problem is apparently that the PAT causes port shifting on the outbound traffic. The router then cannot match the replies up, apparently due to the embedded header that is returned. (Debug does show the ICMP replies arriving at the router.) When you shift to NAT without overload, the router apparently matches up the UDP port numbers and happily shows what you'd expect for trace output.

Further experiments for the reader to try: see if this issue arises with PC `tracert`. Then use `Ethereal` to examine the replies to the router to confirm why the trace is not working as expected.

## Reader Question: Jumbo Frames for SAN?

The reader question: the applications / server group wants jumbo frames enabled to improve SAN performance. What are the pros and cons?

My answer: I would want to weigh the pros and cons carefully.

In favor of jumbos, they may indeed improve performance for SAN. Kevin Tolly's articles have certainly supported this theme over the years. Jumbos allow more to be sent in each frame, so packet-building overhead is reduced, by a factor of 4-6.

One counter-argument is that modern NIC adapters allow this work to be offloaded to the NIC. So if your vendor is telling you that jumbo frames will enhance performance, it gives me the feeling they may really be telling you their SAN CPU or NIC is underpowered. Granted, as the bandwidth goes up, building packet headers does indeed become a burden.

Be that as it may, your firm has probably already bought the SAN unit by the time they got around to asking you about jumbos. This is usually a "we bought it, it now doesn't work as well as we thought it would" sort of thing, isn't it?

If you turn on jumbo frame support, you do need to make sure you do it consistently on the path between the servers and the SAN units. You also need to turn it on along alternative paths. The concern is that if a L2 path is used, missing jumbo support means lost frames, inexplicable drops. If L3 is used, missing an interface means the router or multilayer switch will fragment. This is inefficient, and worse, may spin up the CPU.

Thus the con side of the discussion is that troubleshooting jumbo problems can be painful: use trace with a large payload, check all interfaces and switches or switch ports along the path to make sure jumbos are enabled.

Another con: you really want to isolate the VLANs running jumbos. And make sure that other traffic like web management and SNMP goes to another device interface. Otherwise, you'll potentially have jumbo traffic being routed to normal interfaces, with possible fragmentation. Worse, if sent on a purely L2 path, any receiver without jumbo support will have to discard an inbound jumbo frame. To sum this point up: everybody or nobody on a VLAN should be doing jumbos, don't mix or match.

Putting that together, your team needs precise configuration (both servers and network devices) to make jumbos work.

Another negative: some switches don't support jumbos, especially 9000 Bytes. For instance, smaller Cisco switches, other than the 3750. For more information on Cisco switch support for jumbos, see [http://www.cisco.com/en/US/partner/products/hw/switches/ps700/products\\_configuration\\_example09186a008010edab.shtml](http://www.cisco.com/en/US/partner/products/hw/switches/ps700/products_configuration_example09186a008010edab.shtml). Note that enabling jumbos on some switch models requires a reboot.

By the way, don't confuse NAS with SAN here. The difference is that NAS is a generic NFS or Windows file share, whereas SAN writes blocks. Writing larger blocks can in principle help SAN efficiency. With NAS, there is a lot of traffic back and forth recursively entering directories to open files. NAS bottlenecks are generally related to latency or to all the back-and-forth traffic, less to frame size. There is so much file and directory access overhead that I am inclined to think the file IO is not a major fraction of the packet header-building overhead.

From Googling, the article by Phil Dykstra at <http://sd.wareonearth.com/~phil/jumbo.html> is interesting and has a good point about the WAN / Internet2. I don't see great relevance to LAN -- the concern there still is mostly CPU and packet header building. See also <http://www.abilene.iu.edu/JumboMTU.html>.

If you disagree with any of this, please let me know. This particular topic always seems to generate lively discussions, whenever it has come up in a consulting engagement.

## Conclusion

I hope you enjoyed this article, and hope the PIX tips will be helpful to you.

Your comments, questions, and suggestions for future articles are of course welcome! See below to decipher my email address.

---

Dr. Peter J. Welcher (CCIE #1773, CCSI #94014, CCIP) is a Senior Consultant with Chesapeake NetCraftsmen. NetCraftsmen is a high-end consulting firm and Cisco Premier Partner with multiple specializations, dedicated to quality consulting and knowledge transfer. NetCraftsmen has nine CCIE's, with expertise including large network high-availability routing/switching and design, IP Telephony, QoS, MPLS, IPsec VPN, wireless LAN and bridging, network management, security, IP multicast, and other areas. See <http://www.netcraftsmen.net> for more information about NetCraftsmen. Pete's links start at

<http://www.netcraftsmen.net/welcher> . New articles will be posted under the Articles link. Questions, suggestions for articles, etc. can be sent to pjw <at> netcraftsmen <dot> net.

9/11/2006

Copyright (C) 2006 Peter J. Welcher