NAT Bypass, or NAT()0 on PIX and ASA firewalls is a source of confusion for many. If you take an unconfigured PIX, give it IP addresses on the outside, inside and dmz interfaces, udp and tcp traffic travels happily from higher secuity zones to lower security zones and back again. Then you turn on NAT (or NAPT) for traffic entering the inside interface (so they can get to the internet on the outside) like this: (assuming the inside network is 10.0.1.0/24) global (outside) 1 interface
nat (inside) 1 10.0.1.0 255.255.255.0 and suddenly you can't get to the dmz from inside anymore! Why? Because the nat (inside) 1 10.0.1.0 255.255.255.0 statement says that all incoming traffic from 10.0.10/24 is to get NAPTted. The PIX/ASA is to dumb to realise that traffic not going to the outside doesn't need to be NATted! You have 2 choices: Lets assume the DMZ is 172.16.1.0/24 1. global (dmz) 1 interface
2. access-list INSIDE-NAT-BYPASS permit ip 10.0.1.0 255.255.255.0 172.16.1.0 255.255.255.0
 nat (inside) 0 access-list INSIDE-NAT-BYPASS Now choice 1. above will mean that traffic going from the inside network to the dmz will also get NAPTed - picking up the ip address of the dmz interface on the way through. However, from the point of view of the servers in the dmz, this may not be a great thing, because they will see all inside traffic coming from the ip address of the firewall's dmz interface.
A better way is choice 2. Here we define (using an ACL) that traffic travelling from 10.0.1.0/24 (the inside) to 172.16.1.0/24 (the dmz), then use the special nat()0 construct to say that this traffic when hitting the inside interface is NOT to be NATted. This is the "regular" way to stop traffic being NATted if you don't want it to be. It is a little strange to have a "nat" statement to define traffic that is not to be NATted, but there it is - the special case where we use nat()0 Footnote: You may think that it would be logical to simply define traffic that has to be NATted in an ACL and use that ACL in the "nat (inside) 1" statement. You would do this by excluding traffic going to the DMZ, and including everything else. Unfortunately, the ASA does not support deny rules in Policy Nat - as shown in the error message below when I attempted to try this! 3. access-list INSIDE-NAT-TRAFFIC deny ip 10.0.1.0 255.255.255.0 172.16.1.0 255.255.255.0
 access-list INSIDE-NAT-TRAFFIC permit ip 10.0.1.0 255.255.255.0 any
 nat (inside) 1 access-list INSIDE-NAT-TRAFFIC
 ERROR: Deny rules not supported in Policy Nat