strongSwan - Issue #3403

IKEv2 natd false detection

08.04.2020 13:05 - Simonas Tamosaitis

Status:	Feedback		
Priority:	Normal		
Assignee:			
Category:			
Affected version:	5.8.0	**Resolution:**	

Description

Hello,

I'm having problem with natd in IKEv2.

When strongswan is started and host do not have default route, it generate natd hashes for known IPs and try to send them from 0.0.0.0
sending packet: from 0.0.0.0[500] to x.x.x.x[500] (858 bytes)

After default gateway interface started, daemon detect new ip, but do not regenerate natd hashes.
Wed Apr 8 10:48:47 2020 daemon.info ipsec: 13[KNL] x.x.x.x appeared on wwan0
Wed Apr 8 10:49:12 2020 daemon.info ipsec: 15[NET] sending packet: from 0.0.0.0[500] to x.x.x.x[500] (858 bytes)

Host x.x.x.x gets packet and can't find source IP hash in REQUEST, so it thinks that initiating host is behind NAT, but its not.
Wed Apr 8 10:49:13 2020 daemon.info ipsec: 06[IKE] remote host is behind NAT

So one side is sending UDP encapsulated packets other side is sending packets without encapsulation. Both hosts can send DPD and other packets but won't pass traffic beetween subnets.
Both hosts must not be behind NAT.
How can I solve this situation? After system reboot mobile interface come up later than strongswan, so it happen everytime.

My configuration:
conn ok-ok_c
type=tunnel
left=%any
right=x.x.x.x
leftsubnet=192.168.1.0/24
ikelifetime=8h
lifetime=1h
margintime=9m
keyingtries=3
leftauth=psk
rightauth=psk
rightsubnet=192.168.2.0/24
auto=start
leftid=yyy
rightid=xxx
keyexchange=ikev2
esp=aes128-sha256-modp1536
ike=aes128-sha1-modp1536

History

#1 - 08.04.2020 13:21 - Tobias Brunner

- Status changed from New to Feedback

> So one side is sending UDP encapsulated packets other side is sending packets without encapsulation.

I don't see how that's related because that's wrong anyway if a NAT is detected (even if there isn't really a NAT, e.g. also if *forceencaps* is enabled).

> How can I solve this situation? After system reboot mobile interface come up later than strongswan, so it happen everytime.

So perhaps start strongSwan later or don't use *auto=start*.

#2 - 08.04.2020 13:44 - Simonas Tamosaitis

It is related that connection is broken between hosts. NATd fails, because initiating host is sending misleading information to other one.
Don't you think that it is bug?

forceencaps solve this problem, because both hosts then think there is a NAT situation.
To start later is not solution, this happens too if interface goes down and daemon is restarted for any reason.
To keep *forceencaps* on by default not best idea too.

#3 - 08.04.2020 14:23 - Tobias Brunner

OK, I see what the problem is. On the initiator, the NAT detection is based simply on the NAT-D hash received from the responder. That the returned hash might not have been sent originally (causing a NAT to get detected on the server) is not checked (the sent hashes are not cached). There is also no check whether a NAT-D hash for any local IP address is added to the message. Does the client actually send any NAT_DETECTION_SOURCE_IP notifies at all?

I guess if you used a domain name for *right* (perhaps combined with *charon.retry_initiate_interval*) this would work better as the client would not attempt to send a message in the first place if there is no connectivity/source IP.

#4 - 08.04.2020 15:09 - Simonas Tamosaitis

Yes, client send all source IPs detected at strongswan startup in NAT_DETECTION_SOURCE_IP options in Initiator request. In my case there is 7 IPs. Freshly added IP is not included.
Server then fails to find hash of new source IP and NATD assume that client is behind NAT.
Also server send response with correct hashes to client and client know that he is not behind NAT.

This system will go to production, I can trick my own, but can't deploy it to users.

#5 - 08.04.2020 17:17 - Tobias Brunner

> Yes, client send all source IPs detected at strongswan startup in NAT_DETECTION_SOURCE_IP options in Initiator request. In my case there is 7 IPs. Freshly added IP is not included.

I guess there are two options. Either not initiate the connection until the IP/route is there, or cache the sent hashes and compare them with the returned notify (so at least both of them would consider this to be NAT situation).

#6 - 09.04.2020 14:19 - Simonas Tamosaitis

Thank you for your time, I will try to manage it in some way.