

## strongSwan - Issue #1312

### Long "delete connection" when responder is suddenly off (i.e power or network failure)

18.02.2016 11:15 - Jiri Zendulka

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>	Tobias Brunner	
<b>Category:</b>	configuration	
<b>Affected version:</b>	5.3.5	<b>Resolution:</b> No change required
<b>Description</b>		
Hello,		
I have following situation. An IPsec tunnel has been succesfully established beetwen two devices with strongswan (an initiator and a responder). When I turn off the responder and I stop/restart the initiator then initiator waiting for "deleting connection" for some time - I guess for responder's response which never comes because the responder is currently off. Is it possible reduce this time or disable waiting for "delete notification" response. I think that openswan does not wait for this message.		
Thanks.		

## History

### #1 - 18.02.2016 11:20 - Jiri Zendulka

Connections:

ipsec3: fd07:7::e5...fd07:7::ee IKEv2

ipsec3: local: [fd07:7::e5] uses pre-shared key authentication

ipsec3: remote: [fd07:7::ee] uses pre-shared key authentication

ipsec3: child: fd00:2::/64 === fd00:1::/64 TUNNEL

Security Associations (1 up, 0 connecting):

ipsec3<sup>1</sup>: DELETING, fd07:7::e5[fd07:7::e5]...fd07:7::ee[fd07:7::ee]

ipsec3<sup>1</sup>: IKEv2 SPIs: fe5843c80fde9389\_i\* 1d3684a76c78b90d\_r

ipsec3<sup>1</sup>: IKE proposal: AES\_CBC\_128/HMAC\_SHA1\_96/PRF\_HMAC\_SHA1/MODP\_2048

ipsec3<sup>1</sup>: Tasks active: IKE\_DELETE

ipsec3{1}: INSTALLED, TUNNEL, reqid 1, ESP SPIs: ca7b34a9\_i ca9c1d1c\_o

ipsec3{1}: AES\_CBC\_128/HMAC\_SHA1\_96, 3016 bytes\_i (29 pkts, 23s ago), 5408 bytes\_o (52 pkts, 0s ago), rekeying in 42 minutes

ipsec3{1}: fd00:2::/64 === fd00:1::/64

### #2 - 18.02.2016 19:03 - Noel Kuntze

You can tune the retransmission settings and/or use DPD to drop tunnels to dead peers.

The default retransmission settings make charon wait for up to 76 seconds for a timeout.

From the man page about strongswan.conf

#### IKEv2 RETRANSMISSION

Retransmission timeouts in the IKEv2 daemon charon can be configured globally using the three keys listed below:

```
charon.retransmit_base [1.8]
charon.retransmit_timeout [4.0]
charon.retransmit_tries [5]
```

The following algorithm is used to calculate the timeout:

$$\text{relative timeout} = \text{retransmit\_timeout} * \text{retransmit\_base} ^ (n-1)$$

Where n is the current retransmission count.

Using the default values, packets are retransmitted in:

Retransmission	Relative Timeout	Absolute Timeout
1	4s	4s
2	7s	11s
3	13s	24s
4	23s	47s
5	42s	89s

**#3 - 19.02.2016 08:17 - Jiri Zendulka**

Hi Noel,

many thanks for your answer. But I guess that the charon.retransmit settings have an impact to all ikev2 messages so I am not sure if it is good way to tune timeout of "ike delete" messages. I noticed that ikev1 does not have this issue/feature - there is no waiting for ike delete task. Is it possible to disable waiting for ike delete response? DPD is not suitable solution too because the timeout is quite long. I use shell script to control ipsec tunnels and pending ike delete task is an issue for me - when I restarting/stopping ipsec tunnels I launch ipsec down "tunnelx" and then I check if all tunnels are down via stroke status "tunnelx" if they are down I kill stater deamon. The pending ike delete task blocks killing stater daemon...

**#4 - 29.02.2016 10:27 - Tobias Brunner**

*- Status changed from New to Feedback*

Try ipsec stroke down-nb <name>, which does not block and wait until the exchange completes/times out.

**#5 - 02.03.2016 20:28 - Jiri Zendulka**

Hi Tobias,

thanks for your help. Stroke down-nb does not help in my case. I modified my shell script in the end. When I stop ipsec I kill charon even though there is some tunnel pending "deleting state".

**#6 - 03.03.2016 09:39 - Tobias Brunner**

*- Category set to configuration*

*- Status changed from Feedback to Closed*

*- Assignee set to Tobias Brunner*

*- Resolution set to No change required*

When the daemon terminates it does not wait for a response to the deletes it sends for all existing IKE\_SAs. But I'm closing this as you apparently found a solution.