strongSwan - Bug #1293

Tunnels dropped occasionnally after phase 2 rekeying

02.02.2016 07:47 - Lasse Huovinen

Status:	Closed	**Start date:**	02.02.2016
Priority:	Normal	**Due date:**	
Assignee:	Tobias Brunner	**Estimated time:**	0.00 hour
Category:	libcharon		
Target version:	5.5.0		
Affected version:	5.3.5	**Resolution:**	Fixed

Description

Hello,

We are seeing quite strange tunnel dropping issue with strongSwan and Cisco or Juniper devices when the IPsec tunnels are being rekeyed.

Most of the time everything is working fine but sometimes after phase 2 rekeying the rekeyed tunnel gets dropped. In the case the tunnel gets dropped, strongSwan receives CREATE_CHILD_SA request [N(SET_WINSIZE)] and complains "peer initiated rekeying, but a child is half-open". Then strongSwan replies CREATE_CHILD_SA reponse [N(NO_PROP)].

Here is one example, when the tunnel gets dropped with the Cisco device.

```
##
## strongSwan log
##
Feb  1 04:10:29 02[NET] waiting for data on sockets
Feb  1 04:10:29 09[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.3
1.13[500] (476 bytes)
Feb  1 04:10:29 09[ENC] <signal_voice_policy|3560> parsed CREATE_CHILD_SA request 6 [ N(REKEY_SA)
SA No KE TSi TSr ]
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selecting proposal:
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560>   proposal matches
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> received proposals: ESP:AES_CBC_256/HMAC_SHA1_9
6/MODP_2048/NO_EXT_SEQ
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> configured proposals: ESP:AES_CBC_256/HMAC_SHA1
_96/MODP_2048/NO_EXT_SEQ
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selected proposal: ESP:AES_CBC_256/HMAC_SHA1_96
/MODP_2048/NO_EXT_SEQ
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> got SPI c12d9712
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selecting traffic selectors for us:
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560>  config: 1.1.31.18/32, received: 1.1.31.18/32 =
> match: 1.1.31.18/32
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selecting traffic selectors for other:
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560>  config: 10.102.31.0/24, received: 10.102.31.0/
24 => match: 10.102.31.0/24
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   using AES_CBC for encryption
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   using HMAC_SHA1_96 for integrity
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560> adding inbound ESP SA
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   SPI 0xc12d9712, src 1.1.31.9 dst 10.104.31.13
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> adding SAD entry with SPI c12d9712 and reqid {1
8}  (mark 0/0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using encryption algorithm AES_CBC with key s
ize 256
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using integrity algorithm HMAC_SHA1_96 with k
ey size 160
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using replay window of 32 packets
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560> adding outbound ESP SA
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   SPI 0x59c54e59, src 10.104.31.13 dst 1.1.31.9
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> adding SAD entry with SPI 59c54e59 and reqid {1
```

```
8}  (mark 0/0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using encryption algorithm AES_CBC with key s
ize 256
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using integrity algorithm HMAC_SHA1_96 with k
ey size 160
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using replay window of 32 packets
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 1.1.31.18/32 === 10.102.31.0/24 out  (ma
rk 0/0x00000000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 in  (mar
k 0/0x00000000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (ma
rk 0/0x00000000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 1.1.31.18/32 === 10.102.31.0/24 out  (ma
rk 0/0x00000000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> updating policy 1.1.31.18/32 === 10.102.31.0/24
 out  (mark 0/0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 in  (mar
k 0/0x00000000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> updating policy 10.102.31.0/24 === 1.1.31.18/32
 in  (mark 0/0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (ma
rk 0/0x00000000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> updating policy 10.102.31.0/24 === 1.1.31.18/32
 fwd  (mark 0/0x00000000)
Feb  1 04:10:29 09[IKE] <signal_voice_policy|3560> CHILD_SA signal_voice_policy{4386} established
with SPIs c12d9712_i 59c54e59_o and TS 1.1.31.18/32 === 10.102.31.0/24
Feb  1 04:10:29 09[ENC] <signal_voice_policy|3560> generating CREATE_CHILD_SA response 6 [ SA No K
E TSi TSr ]
Feb  1 04:10:29 09[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.3
1.9[500] (476 bytes)
Feb  1 04:10:29 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb  1 04:10:29 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb  1 04:10:29 02[NET] waiting for data on sockets
Feb  1 04:10:29 08[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.3
1.13[500] (76 bytes)
Feb  1 04:10:29 08[ENC] <signal_voice_policy|3560> parsed INFORMATIONAL request 7 [ N(INVAL_SPI) ]
Feb  1 04:10:29 08[ENC] <signal_voice_policy|3560> generating INFORMATIONAL response 7 [ ]
Feb  1 04:10:29 08[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.3
1.9[500] (76 bytes)
Feb  1 04:10:30 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb  1 04:10:30 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb  1 04:10:30 02[NET] waiting for data on sockets

##
## Here strongSwan receives the request to set the window size but
## we get the error... and strongSwan replies with "NO_PROP"
##

Feb  1 04:10:30 06[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.3
1.13[500] (428 bytes)
Feb  1 04:10:30 06[ENC] <signal_voice_policy|3560> parsed CREATE_CHILD_SA request 8 [ SA No KE N(S
ET_WINSIZE) ]
Feb  1 04:10:30 06[IKE] <signal_voice_policy|3560> peer initiated rekeying, but a child is half-op
en
Feb  1 04:10:30 06[ENC] <signal_voice_policy|3560> generating CREATE_CHILD_SA response 8 [ N(NO_PR
OP) ]
Feb  1 04:10:30 06[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.3
1.9[500] (76 bytes)
Feb  1 04:10:30 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb  1 04:10:30 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb  1 04:10:30 02[NET] waiting for data on sockets

##
## Cisco is not happy with the reply and it decides to delete the tunnel.
##

Feb  1 04:10:30 11[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.3
```

```
1.13[500] (76 bytes)
Feb  1 04:10:30 11[ENC] <signal_voice_policy|3560> parsed INFORMATIONAL request 9 [ D ]
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> received DELETE for ESP CHILD_SA with SPI 85d78
273
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> querying SAD entry with SPI cbd2e0fe  (mark 0/0
x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> querying SAD entry with SPI 85d78273  (mark 0/0
x00000000)
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> closing CHILD_SA signal_voice_policy{4383} with
 SPIs cbd2e0fe_i (163940 bytes) 85d78273_o (159124 bytes) and TS 1.1.31.18/32 === 10.102.31.0/24
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> sending DELETE for ESP CHILD_SA with SPI cbd2e0
fe
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> CHILD_SA closed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 1.1.31.18/32 === 10.102.31.0/24
 out  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not remo
ved
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32
 in  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not remo
ved
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32
 fwd  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not remo
ved
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 1.1.31.18/32 === 10.102.31.0/24
 out  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not remo
ved
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32
 in  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not remo
ved
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32
 fwd  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not remo
ved
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting SAD entry with SPI cbd2e0fe  (mark 0/0
x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleted SAD entry with SPI cbd2e0fe (mark 0/0x0
0000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting SAD entry with SPI 85d78273  (mark 0/0
x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleted SAD entry with SPI 85d78273 (mark 0/0x0
0000000)
Feb  1 04:10:30 11[ENC] <signal_voice_policy|3560> generating INFORMATIONAL response 9 [ D ]
Feb  1 04:10:30 11[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.3
1.9[500] (76 bytes)
Feb  1 04:10:30 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb  1 04:10:51 02[NET] received packet: from 1.1.31.9[500] to 10.106.31.13[500]
Feb  1 04:10:51 02[NET] waiting for data on sockets
Feb  1 04:10:51 10[NET] <emnet_policy|3561> received packet: from 1.1.31.9[500] to 10.106.31.13[50
0] (76 bytes)
Feb  1 04:10:51 10[ENC] <emnet_policy|3561> parsed INFORMATIONAL request 2 [ ]
Feb  1 04:10:51 10[ENC] <emnet_policy|3561> generating INFORMATIONAL response 2 [ ]
Feb  1 04:10:51 10[NET] <emnet_policy|3561> sending packet: from 10.106.31.13[500] to 1.1.31.9[500
] (76 bytes)
Feb  1 04:10:51 04[NET] sending packet: from 10.106.31.13[500] to 1.1.31.9[500]
Feb  1 04:11:25 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb  1 04:11:25 02[NET] waiting for data on sockets
Feb  1 04:11:25 05[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.3
1.13[500] (76 bytes)
Feb  1 04:11:25 05[ENC] <signal_voice_policy|3560> parsed INFORMATIONAL request 10 [ D ]
Feb  1 04:11:25 05[IKE] <signal_voice_policy|3560> received DELETE for IKE_SA signal_voice_policy[
3560]
Feb  1 04:11:25 05[IKE] <signal_voice_policy|3560> deleting IKE_SA signal_voice_policy[3560] betwe
en 10.104.31.13[10.104.31.13]...1.1.31.9[1.1.31.9]
```

```
Feb  1 04:11:25 05[IKE] <signal_voice_policy|3560> IKE_SA signal_voice_policy[3560] state change:
ESTABLISHED => DELETING
Feb  1 04:11:25 05[IKE] <signal_voice_policy|3560> restarting CHILD_SA signal_voice_policy

##
## Since we have enabled
##   closeaction=restart
## in the ipsec.conf file, strongSwan recovers the tunnels but nevertheless
## there is an unfortunate break in the traffic.
##

##
##  From the Cisco point of view, the situation looks like below (the clocks
##  were not synced, this is the same situation as shown above).
##

*Feb  1 01:50:54.250: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF
i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 8
IKEv2 CREATE_CHILD_SA Exchange REQUEST
Payload contents:
 ENCR

*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF
 i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 8
IKEv2 CREATE_CHILD_SA Exchange RESPONSE
Payload contents:
 NOTIFY(NO_PROPOSAL_CHOSEN)

*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Processing any notify-messages in child SA exchange
*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Create child exchange failed

*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):
*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):IKE SA rekey failed
*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Abort exchange
*Feb  1 01:50:54.306: IKEv2:(SA ID = 3):Deleting SA

*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF
i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 9
IKEv2 INFORMATIONAL Exchange REQUEST
Payload contents:
 ENCR
```

We have seen the issue with strongSwan versions 5.3.0 and 5.3.5, and as already mentioed it happens with Cisco and Juniper devices. Since the same issue happens with devices from two different vendors we suspecting there could be something wrong on the strongSwan side. What do you think?

The ipsec.conf is as below. The lifetime values are intentionally so short that we can repeat the problem easily.

We can provide logs from longer period of time, or generate more detailed logs, if that helps to solve the issue. Just let us know what logging levels would be useful.

```
## strongSwan.conf ##

config setup
   uniqueids=never

conn %default
   authby=secret
   rekey=no
   rekeymargin=30s
   rekeyfuzz=0%
```

```
   mobike=no
   keyingtries=%forever
   keyexchange=ikev2
   ike=aes256-sha1-modp2048!
   esp=aes256-sha1-modp2048!
   ikelifetime=240
   lifebytes=67107840
   lifetime=180

conn signal_voice_policy
   type=tunnel
   left=10.104.31.13
   leftsubnet=1.1.31.18/32
   right=1.1.31.9
   rightsubnet=10.102.31.0/24
   closeaction=restart
   auto=start

conn emnet_policy
   type=tunnel
   left=10.106.31.13
   leftsubnet=1.1.31.22/32
   right=1.1.31.9
   rightsubnet=0.0.0.0/0
   closeaction=restart
   auto=start

conn ssh_policy
   type=passthrough
   left=10.106.31.13
   leftsubnet=10.106.31.13/32
   rightsubnet=0.0.0.0/0
   rightprotoport=tcp/ssh
   authby=never
   auto=route

conn ssh_policy2
   type=passthrough
   left=10.106.31.13
   leftsubnet=10.106.31.13/32
   leftprotoport=tcp/ssh
   rightsubnet=0.0.0.0/0
   authby=never
   auto=route
```

Br, Lasse

Associated revisions

Revision 95a5806a - 17.06.2016 18:53 - Tobias Brunner

Merge branch 'exchange-collisions'

Improves the handling of IKEv2 exchange collisions in several corner
cases. TEMPORARY_FAILURE and CHILD_SA_NOT_FOUND notifies that were defined
with RFC 7296 are now handled and sent as appropriate.

The behavior in these situations is tested with new unit tests.

Fixes #379, #464, #876, #1293.

History

#1 - 02.02.2016 15:49 - Tobias Brunner

- Description updated

- Status changed from New to Feedback

This is not really related to the SET_WINDOW_SIZE notify, but rather due to an exchange collision.

Here strongSwan receives the request to set the window size

While such a notify is contained in that request it is actually to rekey the IKE_SA and only indirectly to set the window size (you can also see that in the Cisco log). strongSwan does not support window sizes > 1 but that's not reason why this fails. It fails because there is a CHILD_SA rekeying ongoing at the same time and strongSwan blocks the IKE_SA rekeying request with an error message. The CHILD_SA rekeying is initiated by the Cisco box and strongSwan responds to it but instead of deleting the old CHILD_SA and completing the rekeying the Cisco box sends this:

```
Feb  1 04:10:29 08[ENC] <signal_voice_policy|3560> parsed INFORMATIONAL request 7 [ N(INVAL_SPI) ]
```

The Cisco box sends an INVALID_SPI notify for some reason. I wonder what that's about. Do you seen anything about this in the Cisco log? It also does not seem to attempt to delete the old CHILD_SA. Do you see any messages regarding the CHILD_SA rekeying?

One issue with the above is that strongSwan does not change the state of the old CHILD_SA. So even after the replacement is established successfully it remains in state CHILD_REKEYING, which subsequently blocks the IKE_SA rekeying. But even if we'd e.g. switch the state to CHILD_REKEYED (we use that for IKEv1 only, currently) I wonder if the Cisco box would delete the old SA after rekeying the IKE_SA or if it would just remain installed until it expires (or the IKE_SA is terminated).

Another thing is that according to RFC 7296, section 2.25 a more reasonable response to such an exchange collision would be to return a TEMPORARY_FAILURE notify, which strongSwan currently does not support (was added quite a while after the original code based on RFC 4306 was written). I also don't know if Cisco or Juniper support that error notify or if it would still result in them terminating the IKE_SA.

#2 - 04.02.2016 10:42 - Lasse Huovinen

Thanks for your swift reply! Based on your feedback we did some more further investigations.

Seems that in unsuccessful cases the phase 1 rekeying is initiated by Cisco and Juniper during the phase 2 rekeying (at least from the strongSwan point of view). In the unsuccessfull case Cisco box does not send DELETE (or anything else) to complete phase 2 rekeying. Instead, it sends CREATE_CHILD_SA to rekey phase 1 SAs. Our interpretation is that the Juniper box behaves the same way although it is not so clearly visible in the Juniper logs.

In successful cases the both Cisco and Juniper send DELETE to complete the phase 2 rekeying and everything works fine if the phase 1 rekeying takes place some time after phase 2 was completed.

Is it possible and allowed according to the IKE RFCs that the strongSwan side should consider phase 2 completed if the peer (Cisco/Juniper) initiates phase 1 rekeying meanwhile? Based the logs looks like Cisco and Juniper assume they can do that.

If we increase the difference between phase 1 and phase 2 lifetimes the system becomes more stable, the tunnels stay up as collision will not happen so often. That's no surpise but in the long run the collision will likely still happen. In that sense it would be nice to have some solution for the collision case, too.

```
But even if we'd e.g. switch the state to CHILD_REKEYED (we use that for IKEv1 only, currently) I wonder if th
e Cisco box would delete the old SA after rekeying the IKE_SA or if it would just remain installed until it ex
pires (or the IKE_SA is terminated).
```

We don't know but our guess is "no, Cisco would not likely delete them explicitly". However, would it be worth of trying to make strongSwan IKEv2 to behave the same way as IKEv1? What do you think?

```
The Cisco box sends an INVALID_SPI notify for some reason. I wonder what that's about.
```

What comes to the INVALID_SPI message, we are seeing it during every successful and unsuccessful phase 2 rekeying. Our conclusion is that somehow we manage to send one data packet with an invalid SPI which is not anymore/yet valid from the Cisco point of view. This seems not to be any problem as it happens really for every phase 2 rekeying.

```
Another thing is that according to RFC 7296, section 2.25 a more reasonable response to such an exchange colli
sion would be to return a TEMPORARY_FAILURE notify, which strongSwan currently does not support (was added qui
te a while after the original code based on RFC 4306 was written). I also don't know if Cisco or Juniper suppo
rt that error notify or if it would still result in them terminating the IKE_SA.
```

So far we have no answer to this question.

Here are a few more log snippets that show what happens. The first snippers are from successful phase 2 rekeying followed by phase 1 rekeying. The both strongSwan and Cisco logs are available.

```
Jan 31 12:48:03 02[NET] waiting for data on sockets
Jan 31 12:48:03 08[NET] <signal_voice_policy|2944> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (4
76 bytes)
Jan 31 12:48:03 08[ENC] <signal_voice_policy|2944> parsed CREATE_CHILD_SA request 4 [ N(REKEY_SA) SA No KE TSi
 TSr ]
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944> selecting proposal:
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944>   proposal matches
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944> received proposals: ESP:AES_CBC_256/HMAC_SHA1_96/MODP_2048/
NO_EXT_SEQ
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944> configured proposals: ESP:AES_CBC_256/HMAC_SHA1_96/MODP_204
8/NO_EXT_SEQ
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944> selected proposal: ESP:AES_CBC_256/HMAC_SHA1_96/MODP_2048/N
O_EXT_SEQ
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> got SPI cdfbb6de
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944> selecting traffic selectors for us:
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944>  config: 1.1.31.18/32, received: 1.1.31.18/32 => match: 1.1
.31.18/32
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944> selecting traffic selectors for other:
Jan 31 12:48:03 08[CFG] <signal_voice_policy|2944>  config: 10.102.31.0/24, received: 10.102.31.0/24 => match:
 10.102.31.0/24
Jan 31 12:48:03 08[CHD] <signal_voice_policy|2944>   using AES_CBC for encryption
Jan 31 12:48:03 08[CHD] <signal_voice_policy|2944>   using HMAC_SHA1_96 for integrity
Jan 31 12:48:03 08[CHD] <signal_voice_policy|2944> adding inbound ESP SA
Jan 31 12:48:03 08[CHD] <signal_voice_policy|2944>   SPI 0xcdfbb6de, src 1.1.31.9 dst 10.104.31.13
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> adding SAD entry with SPI cdfbb6de and reqid {18}  (mark 0/
0x00000000)
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944>   using encryption algorithm AES_CBC with key size 256
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944>   using integrity algorithm HMAC_SHA1_96 with key size 160
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944>   using replay window of 32 packets
Jan 31 12:48:03 08[CHD] <signal_voice_policy|2944> adding outbound ESP SA
Jan 31 12:48:03 08[CHD] <signal_voice_policy|2944>   SPI 0x46f11ed1, src 10.104.31.13 dst 1.1.31.9
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> adding SAD entry with SPI 46f11ed1 and reqid {18}  (mark 0/
0x00000000)
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944>   using encryption algorithm AES_CBC with key size 256
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944>   using integrity algorithm HMAC_SHA1_96 with key size 160
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944>   using replay window of 32 packets
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark 0/0x00000
000) already exists, increasing refcount
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0/0x000000
00) already exists, increasing refcount
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark 0/0x00000
000) already exists, increasing refcount
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark 0/0x00000
000) already exists, increasing refcount
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> updating policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark
0/0x00000000)
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0/0x000000
00) already exists, increasing refcount
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> updating policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0
/0x00000000)
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark 0/0x00000
000) already exists, increasing refcount
Jan 31 12:48:03 08[KNL] <signal_voice_policy|2944> updating policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark
0/0x00000000)
Jan 31 12:48:03 08[IKE] <signal_voice_policy|2944> CHILD_SA signal_voice_policy{3630} established with SPIs cd
fbb6de_i 46f11ed1_o and TS 1.1.31.18/32 === 10.102.31.0/24
##
##  strongSwan replies to phase 2 rekeying...
##
Jan 31 12:48:03 08[ENC] <signal_voice_policy|2944> generating CREATE_CHILD_SA response 4 [ SA No KE TSi TSr ]
Jan 31 12:48:03 08[NET] <signal_voice_policy|2944> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (47
6 bytes)
Jan 31 12:48:03 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Jan 31 12:48:03 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Jan 31 12:48:03 02[NET] waiting for data on sockets
Jan 31 12:48:03 10[NET] <signal_voice_policy|2944> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (7
6 bytes)
##
##  Here we see the INVALID_SPI message from Cisco, but it has no negative
##  effect to rekeying process.
##
Jan 31 12:48:03 10[ENC] <signal_voice_policy|2944> parsed INFORMATIONAL request 5 [ N(INVAL_SPI) ]
Jan 31 12:48:03 10[ENC] <signal_voice_policy|2944> generating INFORMATIONAL response 5 [ ]
Jan 31 12:48:03 10[NET] <signal_voice_policy|2944> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76
 bytes)
```

```
Jan 31 12:48:03 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Jan 31 12:48:03 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Jan 31 12:48:03 02[NET] waiting for data on sockets
Jan 31 12:48:03 15[NET] <signal_voice_policy|2944> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (7
6 bytes)
##
##  strongSwan receives delete for the expired phase 2 SAs from Cisco and
##  they are deleted.
##
Jan 31 12:48:03 15[ENC] <signal_voice_policy|2944> parsed INFORMATIONAL request 6 [ D ]
Jan 31 12:48:03 15[IKE] <signal_voice_policy|2944> received DELETE for ESP CHILD_SA with SPI 41dd6805
Jan 31 12:48:03 15[KNL] <signal_voice_policy|2944> querying SAD entry with SPI c487739c  (mark 0/0x00000000)
Jan 31 12:48:03 15[KNL] <signal_voice_policy|2944> querying SAD entry with SPI 41dd6805  (mark 0/0x00000000)
Jan 31 12:48:03 15[IKE] <signal_voice_policy|2944> closing CHILD_SA signal_voice_policy{3628} with SPIs c48773
9c_i (180536 bytes) 41dd6805_o (174860 bytes) and TS 1.1.31.18/32 === 10.102.31.0/24
Jan 31 12:48:03 15[IKE] <signal_voice_policy|2944> sending DELETE for ESP CHILD_SA with SPI c487739c
Jan 31 12:48:03 15[IKE] <signal_voice_policy|2944> CHILD_SA closed
Jan 31 12:48:03 15[KNL] <signal_voice_policy|2944> deleting policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark
0/0x00000000)
Jan 31 12:48:03 15[KNL] <signal_voice_policy|2944> policy still used by another CHILD_SA, not removed
Jan 31 12:48:03 15[KNL] <signal_voice_policy|2944> deleting policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0
/0x00000000)
Jan 31 12:48:03 15[KNL] <signal_voice_policy|2944> policy still used by another CHILD_SA, not removed
Jan 31 12:48:03 15[KNL] <signal_voice_policy|2944> deleting policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark
0/0x00000000)
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> policy still used by another CHILD_SA, not removed
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> deleting policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark
0/0x00000000)
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> policy still used by another CHILD_SA, not removed
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> deleting policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0
/0x00000000)
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> policy still used by another CHILD_SA, not removed
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> deleting policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark
0/0x00000000)
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> policy still used by another CHILD_SA, not removed
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> deleting SAD entry with SPI c487739c  (mark 0/0x00000000)
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> deleted SAD entry with SPI c487739c (mark 0/0x00000000)
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> deleting SAD entry with SPI 41dd6805  (mark 0/0x00000000)
Jan 31 12:48:04 15[KNL] <signal_voice_policy|2944> deleted SAD entry with SPI 41dd6805 (mark 0/0x00000000)
Jan 31 12:48:04 15[ENC] <signal_voice_policy|2944> generating INFORMATIONAL response 6 [ D ]
Jan 31 12:48:04 15[NET] <signal_voice_policy|2944> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76
 bytes)
Jan 31 12:48:04 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Jan 31 12:48:15 02[NET] received packet: from 1.1.31.9[500] to 10.106.31.13[500]
Jan 31 12:48:15 02[NET] waiting for data on sockets
Jan 31 12:48:15 05[NET] <emnet_policy|2945> received packet: from 1.1.31.9[500] to 10.106.31.13[500] (76 bytes
)
Jan 31 12:48:15 05[ENC] <emnet_policy|2945> parsed INFORMATIONAL request 3 [ ]
Jan 31 12:48:15 05[ENC] <emnet_policy|2945> generating INFORMATIONAL response 3 [ ]
Jan 31 12:48:15 05[NET] <emnet_policy|2945> sending packet: from 10.106.31.13[500] to 1.1.31.9[500] (76 bytes)
Jan 31 12:48:15 04[NET] sending packet: from 10.106.31.13[500] to 1.1.31.9[500]
Jan 31 12:48:30 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Jan 31 12:48:30 02[NET] waiting for data on sockets
Jan 31 12:48:30 14[NET] <signal_voice_policy|2944> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (4
28 bytes)
##
##  The phase 1 rekeying has been initiated by the Cisco box and
##  it takes place some 26 secs after the previous phase 2 rekeying was completed.
##
Jan 31 12:48:30 14[ENC] <signal_voice_policy|2944> parsed CREATE_CHILD_SA request 7 [ SA No KE N(SET_WINSIZE)
]
Jan 31 12:48:30 14[IKE] <signal_voice_policy|2944> 1.1.31.9 is initiating an IKE_SA
Jan 31 12:48:30 14[IKE] <signal_voice_policy|2944> IKE_SA signal_voice_policy[2946] state change: CREATED => C
ONNECTING
Jan 31 12:48:30 14[CFG] <signal_voice_policy|2944> selecting proposal:
Jan 31 12:48:30 14[CFG] <signal_voice_policy|2944>   proposal matches
Jan 31 12:48:30 14[CFG] <signal_voice_policy|2944> received proposals: IKE:AES_CBC_256/HMAC_SHA1_96/PRF_HMAC_S
HA1/MODP_2048
Jan 31 12:48:30 14[CFG] <signal_voice_policy|2944> configured proposals: IKE:AES_CBC_256/HMAC_SHA1_96/PRF_HMAC
_SHA1/MODP_2048
Jan 31 12:48:30 14[CFG] <signal_voice_policy|2944> selected proposal: IKE:AES_CBC_256/HMAC_SHA1_96/PRF_HMAC_SH
A1/MODP_2048
Jan 31 12:48:30 14[IKE] <signal_voice_policy|2944> IKE_SA signal_voice_policy[2944] state change: ESTABLISHED
=> REKEYING
Jan 31 12:48:30 14[ENC] <signal_voice_policy|2944> generating CREATE_CHILD_SA response 7 [ SA No KE ]
```

```
Jan 31 12:48:30 14[NET] <signal_voice_policy|2944> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (42
8 bytes)
Jan 31 12:48:30 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Jan 31 12:48:30 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Jan 31 12:48:30 02[NET] waiting for data on sockets
Jan 31 12:48:30 16[NET] <signal_voice_policy|2944> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (7
6 bytes)
Jan 31 12:48:30 16[ENC] <signal_voice_policy|2944> parsed INFORMATIONAL request 8 [ D ]
Jan 31 12:48:30 16[IKE] <signal_voice_policy|2944> IKE_SA signal_voice_policy[2946] state change: CONNECTING =
> ESTABLISHED
Jan 31 12:48:30 16[IKE] <signal_voice_policy|2944> IKE_SA signal_voice_policy[2946] rekeyed between 10.104.31.
13[10.104.31.13]...1.1.31.9[1.1.31.9]
Jan 31 12:48:30 16[IKE] <signal_voice_policy|2944> received DELETE for IKE_SA signal_voice_policy[2944]
Jan 31 12:48:31 16[IKE] <signal_voice_policy|2944> deleting IKE_SA signal_voice_policy[2944] between 10.104.31
.13[10.104.31.13]...1.1.31.9[1.1.31.9]
Jan 31 12:48:31 16[IKE] <signal_voice_policy|2944> IKE_SA signal_voice_policy[2944] state change: REKEYING =>
DELETING
Jan 31 12:48:31 16[IKE] <signal_voice_policy|2944> IKE_SA deleted
Jan 31 12:48:31 16[ENC] <signal_voice_policy|2944> generating INFORMATIONAL response 8 [ ]
Jan 31 12:48:31 16[NET] <signal_voice_policy|2944> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76
 bytes)
Jan 31 12:48:31 16[IKE] <signal_voice_policy|2944> IKE_SA signal_voice_policy[2944] state change: DELETING =>
DESTROYING
```

Successful case as seen in Cisco log.

```
*Jan 31 18:28:32.708: IPSEC(lifetime_expiry): SA lifetime threshold reached, expiring in 30 seconds
*Jan 31 18:28:32.708: IPSEC(sa_request): ,
  (key eng. msg.) OUTBOUND local= 1.1.31.9:500, remote= 10.104.31.13:500,
    local_proxy= 10.102.31.0/255.255.255.0/256/0,
    remote_proxy= 1.1.31.18/255.255.255.255/256/0,
    protocol= ESP, transform= esp-aes 256 esp-sha-hmac  (Tunnel),
    lifedur= 180s and 10000kb,
    spi= 0xC1FD0178(3254583672), conn_id= 0, keysize= 256, flags= 0x0
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Check for IPSEC rekey
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Beginning IPSec Rekey as Initiator
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Set IPSEC DH group
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Checking for PFS configuration
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):PFS configured, DH group 14
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):[IKEv2 -> Crypto Engine] Computing DH public key, DH Group 14
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):[Crypto Engine -> IKEv2] DH key Computation PASSED
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Request queued for computation of DH key
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Generating CREATE_CHILD_SA exchange
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):ESP Proposal: 1, SPI size: 4 (IPSec negotiation),
Num. transforms: 4
   AES-CBC   SHA96   DH_GROUP_2048_MODP/Group 14   Don't use ESN
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Building packet for encryption.
Payload contents:
 NOTIFY(REKEY_SA) SA N KE TSi TSr
*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Checking if request will fit in peer window

*Jan 31 18:28:32.708: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : 656B51C56E613646 - Responder SPI : 74A50CE879DC768D Message id: 0
IKEv2 CREATE_CHILD_SA Exchange REQUEST
Payload contents:
 ENCR

*Jan 31 18:28:32.948: %CRYPTO-4-RECVD_PKT_INV_SPI: decaps: rec'd IPSEC packet has invalid spi for destaddr=1.1
.31.9, prot
                =50, spi=0xED95BD74(3986013556), srcaddr=10.104.31.13, input interface=Vlan309
*Jan 31 18:28:32.952: IKEv2:(SA ID = 2):Sending INVALID_SPI notify
*Jan 31 18:28:32.952: IKEv2:(SA ID = 2):Building packet for encryption.
Payload contents:
 NOTIFY(INVALID_SPI)
##
##  The INVALID_SPI notification seen also in successful case.
##
*Jan 31 18:28:32.952: IKEv2:(SA ID = 2):Checking if request will fit in peer window
*Jan 31 18:28:32.952: IKEv2:(SA ID = 2):Error encountered while navigating State Machine

*Jan 31 18:28:32.952: IKEv2:(SA ID = 2):No Result Transition table avail for INFO_I_BLD_INFO / EV_SEND_INVALID
_SPI with r
                eturn code 0.0.0.36

*Jan 31 18:28:32.976: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
```

```
Initiator SPI : 656B51C56E613646 - Responder SPI : 74A50CE879DC768D Message id: 0
IKEv2 CREATE_CHILD_SA Exchange RESPONSE
Payload contents:
 SA N KE TSi TSr

###
###  In this successful case Cisco does not need to initiate IKE SA rekeying as
###  compared to the unsuccessful case, see below.
###

*Jan 31 18:28:32.976: IKEv2:(SA ID = 2):Processing any notify-messages in child SA exchange
*Jan 31 18:28:32.976: IKEv2:(SA ID = 2):Validating create child message
*Jan 31 18:28:32.976: IKEv2:(SA ID = 2):Processing CREATE_CHILD_SA exchange
*Jan 31 18:28:32.976: IKEv2:KMI/verify policy/sending to IPSec:
         prot: 3 txfm: 12 hmac 2 flags 129 keysize 256 IDB 0xF17B58C
*Jan 31 18:28:32.976: IPSEC(validate_proposal_request): proposal part #1
*Jan 31 18:28:32.976: IPSEC(validate_proposal_request): proposal part #1,
  (key eng. msg.) INBOUND local= 1.1.31.9:0, remote= 10.104.31.13:0,
    local_proxy= 10.102.31.0/255.255.255.0/256/0,
    remote_proxy= 1.1.31.18/255.255.255.255/256/0,
    protocol= ESP, transform= NONE  (Tunnel),
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize= 256, flags= 0x0
*Jan 31 18:28:32.980: Crypto mapdb : proxy_match
        src addr     : 10.102.31.0
        dst addr     : 1.1.31.18
        protocol     : 0
        src port     : 0
        dst port     : 0
*Jan 31 18:28:32.980: IKEv2:(SA ID = 2):Checking for PFS configuration
*Jan 31 18:28:32.980: IKEv2:(SA ID = 2):PFS configured, DH group 14
*Jan 31 18:28:32.980: IKEv2:(SA ID = 2):[IKEv2 -> Crypto Engine] Computing DH secret key, DH Group 14
*Jan 31 18:28:32.980: IKEv2:(SA ID = 2):Request queued for computation of DH secret
*Jan 31 18:28:32.996: IKEv2:(SA ID = 2):[Crypto Engine -> IKEv2] DH key Computation PASSED
*Jan 31 18:28:32.996: IKEv2:(SA ID = 2):Checking if IKE SA rekey
*Jan 31 18:28:32.996: IKEv2:(SA ID = 2):Load IPSEC key material
*Jan 31 18:28:32.996: IKEv2:(SA ID = 2):[IKEv2 -> IPsec] Create IPsec SA into IPsec database
*Jan 31 18:28:32.996: IKEv2:(SA ID = 2):Asynchronous request queued

*Jan 31 18:28:32.996: IKEv2:(SA ID = 2):
*Jan 31 18:28:32.996: IPSEC(key_engine): got a queue event with 1 KMI message(s)
*Jan 31 18:28:32.996: Crypto mapdb : proxy_match
        src addr     : 10.102.31.0
        dst addr     : 1.1.31.18
        protocol     : 256
        src port     : 0
        dst port     : 0
*Jan 31 18:28:32.996: IPSEC(crypto_ipsec_create_ipsec_sas): Map found Virtual-Access1-head-0
*Jan 31 18:28:32.996: IPSEC(create_sa): sa created,
  (sa) sa_dest= 1.1.31.9, sa_proto= 50,
    sa_spi= 0xED95BD74(3986013556),
    sa_trans= esp-aes 256 esp-sha-hmac , sa_conn_id= 216
    sa_lifetime(k/sec)= (10000/180)
*Jan 31 18:28:32.996: IPSEC(create_sa): sa created,
  (sa) sa_dest= 10.104.31.13, sa_proto= 50,
    sa_spi= 0xCE6B7FDB(3463151579),
    sa_trans= esp-aes 256 esp-sha-hmac , sa_conn_id= 215
    sa_lifetime(k/sec)= (10000/180)
*Jan 31 18:28:33.000: IPSEC(update_current_outbound_sa): updated peer 10.104.31.13 current outbound sa to SPI
CE6B7FDB
*Jan 31 18:28:33.000: IPSEC: Expand action denied, notify RP
*Jan 31 18:28:33.000: IPSEC(rte_mgr): VPN Route Event Install new outbound sa: Create IPV4 route from ACL for
10.104.31.1
                 3
*Jan 31 18:28:33.000: IPSEC(rte_mgr): VPN Route Refcount 2 10.104.31.13 on Virtual-Access1
*Jan 31 18:28:33.000: IKEv2:(SA ID = 2):[IPsec -> IKEv2] Creation of IPsec SA into IPsec database PASSED
*Jan 31 18:28:33.000: IKEv2:(SA ID = 2):IKEV2 SA created; inserting SA into database. SA lifetime timer (240 s
ec) started

*Jan 31 18:28:33.000: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : 656B51C56E613646 - Responder SPI : 74A50CE879DC768D Message id: 1
IKEv2 INFORMATIONAL Exchange REQUEST
Payload contents:
 ENCR
```

```
*Jan 31 18:28:33.000: IKEv2:(SA ID = 2):Sending DELETE INFO message for IPsec SA [SPI: 0xC1FD0178]
*Jan 31 18:28:33.000: IKEv2:(SA ID = 2):Building packet for encryption.
Payload contents:
 DELETE
*Jan 31 18:28:33.000: IKEv2:(SA ID = 2):Checking if request will fit in peer window
*Jan 31 18:28:33.000: IKEv2:(SA ID = 2):Check for existing IPSEC SA

*Jan 31 18:28:33.056: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : 656B51C56E613646 - Responder SPI : 74A50CE879DC768D Message id: 1
IKEv2 INFORMATIONAL Exchange RESPONSE
Payload contents:

*Jan 31 18:28:33.060: IKEv2:(SA ID = 2):Processing ACK to informational exchange

*Jan 31 18:28:33.060: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : 656B51C56E613646 - Responder SPI : 74A50CE879DC768D Message id: 2
IKEv2 INFORMATIONAL Exchange REQUEST
Payload contents:
 ENCR

*Jan 31 18:28:33.172: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : 656B51C56E613646 - Responder SPI : 74A50CE879DC768D Message id: 2
IKEv2 INFORMATIONAL Exchange RESPONSE
Payload contents:
 DELETE
```

For comparison, the unsuccessful case. The first log snipper is from strongSwan
and the second one from Cisco.

```
Feb  1 04:10:29 02[NET] waiting for data on sockets
Feb  1 04:10:29 09[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (4
76 bytes)
Feb  1 04:10:29 09[ENC] <signal_voice_policy|3560> parsed CREATE_CHILD_SA request 6 [ N(REKEY_SA) SA No KE TSi
 TSr ]
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selecting proposal:
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560>   proposal matches
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> received proposals: ESP:AES_CBC_256/HMAC_SHA1_96/MODP_2048/
NO_EXT_SEQ
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> configured proposals: ESP:AES_CBC_256/HMAC_SHA1_96/MODP_204
8/NO_EXT_SEQ
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selected proposal: ESP:AES_CBC_256/HMAC_SHA1_96/MODP_2048/N
O_EXT_SEQ
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> got SPI c12d9712
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selecting traffic selectors for us:
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560>  config: 1.1.31.18/32, received: 1.1.31.18/32 => match: 1.1
.31.18/32
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560> selecting traffic selectors for other:
Feb  1 04:10:29 09[CFG] <signal_voice_policy|3560>  config: 10.102.31.0/24, received: 10.102.31.0/24 => match:
 10.102.31.0/24
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   using AES_CBC for encryption
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   using HMAC_SHA1_96 for integrity
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560> adding inbound ESP SA
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   SPI 0xc12d9712, src 1.1.31.9 dst 10.104.31.13
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> adding SAD entry with SPI c12d9712 and reqid {18}  (mark 0/
0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using encryption algorithm AES_CBC with key size 256
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using integrity algorithm HMAC_SHA1_96 with key size 160
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using replay window of 32 packets
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560> adding outbound ESP SA
Feb  1 04:10:29 09[CHD] <signal_voice_policy|3560>   SPI 0x59c54e59, src 10.104.31.13 dst 1.1.31.9
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> adding SAD entry with SPI 59c54e59 and reqid {18}  (mark 0/
0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using encryption algorithm AES_CBC with key size 256
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using integrity algorithm HMAC_SHA1_96 with key size 160
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560>   using replay window of 32 packets
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark 0/0x00000
000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0/0x000000
00) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark 0/0x00000
000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark 0/0x00000
000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> updating policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark
```

```
0/0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0/0x000000
00) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> updating policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0
/0x00000000)
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark 0/0x00000
000) already exists, increasing refcount
Feb  1 04:10:29 09[KNL] <signal_voice_policy|3560> updating policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark
0/0x00000000)
Feb  1 04:10:29 09[IKE] <signal_voice_policy|3560> CHILD_SA signal_voice_policy{4386} established with SPIs c1
2d9712_i 59c54e59_o and TS 1.1.31.18/32 === 10.102.31.0/24
Feb  1 04:10:29 09[ENC] <signal_voice_policy|3560> generating CREATE_CHILD_SA response 6 [ SA No KE TSi TSr ]
Feb  1 04:10:29 09[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (47
6 bytes)
Feb  1 04:10:29 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb  1 04:10:29 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb  1 04:10:29 02[NET] waiting for data on sockets
Feb  1 04:10:29 08[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (7
6 bytes)
Feb  1 04:10:29 08[ENC] <signal_voice_policy|3560> parsed INFORMATIONAL request 7 [ N(INVAL_SPI) ]
Feb  1 04:10:29 08[ENC] <signal_voice_policy|3560> generating INFORMATIONAL response 7 [ ]
Feb  1 04:10:29 08[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76
 bytes)

###
###  Instead of DELETE, strongSwan receives the request to rekey the
###  phase 1 SA and then we're in trouble, as discussed already...
###

Feb  1 04:10:30 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb  1 04:10:30 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb  1 04:10:30 02[NET] waiting for data on sockets
Feb  1 04:10:30 06[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (4
28 bytes)
Feb  1 04:10:30 06[ENC] <signal_voice_policy|3560> parsed CREATE_CHILD_SA request 8 [ SA No KE N(SET_WINSIZE)
]
Feb  1 04:10:30 06[IKE] <signal_voice_policy|3560> peer initiated rekeying, but a child is half-open
Feb  1 04:10:30 06[ENC] <signal_voice_policy|3560> generating CREATE_CHILD_SA response 8 [ N(NO_PROP) ]
Feb  1 04:10:30 06[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76
 bytes)
Feb  1 04:10:30 04[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb  1 04:10:30 02[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb  1 04:10:30 02[NET] waiting for data on sockets
Feb  1 04:10:30 11[NET] <signal_voice_policy|3560> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (7
6 bytes)
Feb  1 04:10:30 11[ENC] <signal_voice_policy|3560> parsed INFORMATIONAL request 9 [ D ]
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> received DELETE for ESP CHILD_SA with SPI 85d78273
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> querying SAD entry with SPI cbd2e0fe  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> querying SAD entry with SPI 85d78273  (mark 0/0x00000000)
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> closing CHILD_SA signal_voice_policy{4383} with SPIs cbd2e0
fe_i (163940 bytes) 85d78273_o (159124 bytes) and TS 1.1.31.18/32 === 10.102.31.0/24
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> sending DELETE for ESP CHILD_SA with SPI cbd2e0fe
Feb  1 04:10:30 11[IKE] <signal_voice_policy|3560> CHILD_SA closed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark
0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not removed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0
/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not removed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark
0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not removed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 1.1.31.18/32 === 10.102.31.0/24 out  (mark
0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not removed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32 in  (mark 0
/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not removed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting policy 10.102.31.0/24 === 1.1.31.18/32 fwd  (mark
0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> policy still used by another CHILD_SA, not removed
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting SAD entry with SPI cbd2e0fe  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleted SAD entry with SPI cbd2e0fe (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleting SAD entry with SPI 85d78273  (mark 0/0x00000000)
Feb  1 04:10:30 11[KNL] <signal_voice_policy|3560> deleted SAD entry with SPI 85d78273 (mark 0/0x00000000)
```

```
Feb  1 04:10:30 11[ENC] <signal_voice_policy|3560> generating INFORMATIONAL response 9 [ D ]
Feb  1 04:10:30 11[NET] <signal_voice_policy|3560> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76
 bytes)

*Feb  1 01:50:53.926: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 6
IKEv2 CREATE_CHILD_SA Exchange REQUEST
Payload contents:
 ENCR

*Feb  1 01:50:54.178: %CRYPTO-4-RECVD_PKT_INV_SPI: decaps: rec'd IPSEC packet has invalid spi for destaddr=1.1
.31.9, prot
                =50, spi=0x59C54E59(1506102873), srcaddr=10.104.31.13, input interface=Vlan309
*Feb  1 01:50:54.182: IKEv2:(SA ID = 2):Sending INVALID_SPI notify
*Feb  1 01:50:54.182: IKEv2:(SA ID = 2):Building packet for encryption.
Payload contents:
 NOTIFY(INVALID_SPI)
*Feb  1 01:50:54.182: IKEv2:(SA ID = 2):Checking if request will fit in peer window
*Feb  1 01:50:54.182: IKEv2:(SA ID = 2):Error encountered while navigating State Machine

*Feb  1 01:50:54.182: IKEv2:(SA ID = 2):No Result Transition table avail for INFO_I_BLD_INFO / EV_SEND_INVALID
_SPI with r
                eturn code 0.0.0.36

###
###   From Cisco point of view the difference between successful and unsuccessful case is visible here!!!
###   Cisco checks need for IKE SA rekeying and yes, in this case the IKE SA needs to be rekeyed.
###   Cisco sends CREATE_CHILD_SA but it does not send DELETE as it does in successful case.
###   As discussed already, strongSwan is not happy to this message, and it replies with NO_PROPOSAL_CHOSEN,
###   see below.
###
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):Checking if we need to rekey the IKE SA
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):Initiating a rekey
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):Beginning IKE Rekey as Initiator
*Feb  1 01:50:54.202: IKEv2:Searching Policy with fvrf 0, local address 1.1.31.9
*Feb  1 01:50:54.202: IKEv2:Found Policy 'TBS_POL'
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):[IKEv2 -> Crypto Engine] Computing DH public key, DH Group 14
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):[Crypto Engine -> IKEv2] DH key Computation PASSED
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):Request queued for computation of DH key
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):Generating CREATE_CHILD_SA exchange
*Feb  1 01:50:54.202: IKEv2:(SA ID = 3):IKE Proposal: 1, SPI size: 8 (rekey),
Num. transforms: 4
   AES-CBC   SHA1   SHA96   DH_GROUP_2048_MODP/Group 14
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):Building packet for encryption.
Payload contents:
 SA N KE NOTIFY(SET_WINDOW_SIZE)
*Feb  1 01:50:54.202: IKEv2:(SA ID = 2):Checking if request will fit in peer window

*Feb  1 01:50:54.210: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 6
IKEv2 CREATE_CHILD_SA Exchange RESPONSE
Payload contents:
 SA N KE TSi TSr

*Feb  1 01:50:54.214: IKEv2:(SA ID = 2):Processing any notify-messages in child SA exchange
*Feb  1 01:50:54.214: IKEv2:(SA ID = 2):Validating create child message
*Feb  1 01:50:54.214: IKEv2:(SA ID = 2):Processing CREATE_CHILD_SA exchange
*Feb  1 01:50:54.214: IKEv2:KMI/verify policy/sending to IPSec:
         prot: 3 txfm: 12 hmac 2 flags 129 keysize 256 IDB 0xF17B58C
*Feb  1 01:50:54.214: IPSEC(validate_proposal_request): proposal part #1
*Feb  1 01:50:54.214: IPSEC(validate_proposal_request): proposal part #1,
  (key eng. msg.) INBOUND local= 1.1.31.9:0, remote= 10.104.31.13:0,
    local_proxy= 10.102.31.0/255.255.255.0/256/0,
    remote_proxy= 1.1.31.18/255.255.255.255/256/0,
    protocol= ESP, transform= NONE  (Tunnel),
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize= 256, flags= 0x0
*Feb  1 01:50:54.214: Crypto mapdb : proxy_match
        src addr     : 10.102.31.0
        dst addr     : 1.1.31.18
        protocol     : 0
        src port     : 0
        dst port     : 0
*Feb  1 01:50:54.214: IKEv2:(SA ID = 2):Checking for PFS configuration
*Feb  1 01:50:54.214: IKEv2:(SA ID = 2):PFS configured, DH group 14
```

```
*Feb  1 01:50:54.214: IKEv2:(SA ID = 2):[IKEv2 -> Crypto Engine] Computing DH secret key, DH Group 14
*Feb  1 01:50:54.214: IKEv2:(SA ID = 2):Request queued for computation of DH secret
*Feb  1 01:50:54.230: IKEv2:(SA ID = 2):[Crypto Engine -> IKEv2] DH key Computation PASSED
*Feb  1 01:50:54.230: IKEv2:(SA ID = 2):Checking if IKE SA rekey
*Feb  1 01:50:54.230: IKEv2:(SA ID = 2):Load IPSEC key material
*Feb  1 01:50:54.230: IKEv2:(SA ID = 2):[IKEv2 -> IPsec] Create IPsec SA into IPsec database
*Feb  1 01:50:54.230: IKEv2:(SA ID = 2):Asynchronous request queued

*Feb  1 01:50:54.230: IKEv2:(SA ID = 2):
*Feb  1 01:50:54.230: IPSEC(key_engine): got a queue event with 1 KMI message(s)
*Feb  1 01:50:54.230: Crypto mapdb : proxy_match
        src addr     : 10.102.31.0
        dst addr     : 1.1.31.18
        protocol     : 256
        src port     : 0
        dst port     : 0
*Feb  1 01:50:54.230: IPSEC(crypto_ipsec_create_ipsec_sas): Map found Virtual-Access1-head-0
*Feb  1 01:50:54.234: IPSEC(create_sa): sa created,
  (sa) sa_dest= 1.1.31.9, sa_proto= 50,
    sa_spi= 0x59C54E59(1506102873),
    sa_trans= esp-aes 256 esp-sha-hmac , sa_conn_id= 940
    sa_lifetime(k/sec)= (10000/180)
*Feb  1 01:50:54.234: IPSEC(create_sa): sa created,
  (sa) sa_dest= 10.104.31.13, sa_proto= 50,
    sa_spi= 0xC12D9712(3240990482),
    sa_trans= esp-aes 256 esp-sha-hmac , sa_conn_id= 939
    sa_lifetime(k/sec)= (10000/180)
*Feb  1 01:50:54.234: IPSEC(update_current_outbound_sa): updated peer 10.104.31.13 current outbound sa to SPI
C12D9712
*Feb  1 01:50:54.234: IPSEC: Expand action denied, notify RP
*Feb  1 01:50:54.234: IPSEC(rte_mgr): VPN Route Event Install new outbound sa: Create IPV4 route from ACL for
10.104.31.1
                3
*Feb  1 01:50:54.234: IPSEC(rte_mgr): VPN Route Refcount 2 10.104.31.13 on Virtual-Access1
*Feb  1 01:50:54.234: IKEv2:(SA ID = 2):[IPsec -> IKEv2] Creation of IPsec SA into IPsec database PASSED
*Feb  1 01:50:54.234: IKEv2:(SA ID = 2):IKEV2 SA created; inserting SA into database. SA lifetime timer (240 s
ec) started

*Feb  1 01:50:54.234: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 7
IKEv2 INFORMATIONAL Exchange REQUEST
Payload contents:
 ENCR

*Feb  1 01:50:54.234: IKEv2:(SA ID = 2):Sending DELETE INFO message for IPsec SA [SPI: 0x85D78273]
*Feb  1 01:50:54.234: IKEv2:(SA ID = 2):Building packet for encryption.
Payload contents:
 DELETE
*Feb  1 01:50:54.238: IKEv2:(SA ID = 2):Checking if request will fit in peer window
*Feb  1 01:50:54.238: IKEv2:(SA ID = 2):Check for existing IPSEC SA

*Feb  1 01:50:54.250: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 7
IKEv2 INFORMATIONAL Exchange RESPONSE
Payload contents:

*Feb  1 01:50:54.250: IKEv2:(SA ID = 2):Processing ACK to informational exchange

*Feb  1 01:50:54.250: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 8
IKEv2 CREATE_CHILD_SA Exchange REQUEST
Payload contents:
 ENCR

###
### Response from strongSwan...
###
*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 8
IKEv2 CREATE_CHILD_SA Exchange RESPONSE
Payload contents:
 NOTIFY(NO_PROPOSAL_CHOSEN)

*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Processing any notify-messages in child SA exchange
```

```
*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Create child exchange failed

*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):
*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):IKE SA rekey failed
*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Abort exchange
*Feb  1 01:50:54.306: IKEv2:(SA ID = 3):Deleting SA

*Feb  1 01:50:54.306: IKEv2:(SA ID = 2):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 9
IKEv2 INFORMATIONAL Exchange REQUEST
Payload contents:
 ENCR

###
###  As a consequence, the Cisco closes the IKE SA....
###
*Feb  1 01:50:54.386: IKEv2:(SA ID = 2):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : BEA75F41D21B0BEB - Responder SPI : 30453E7655BCD79D Message id: 9
IKEv2 INFORMATIONAL Exchange RESPONSE
Payload contents:
 DELETE
```

#3 - 04.02.2016 12:50 - Tobias Brunner

> Is it possible and allowed according to the IKE RFCs that the strongSwan side
> should consider phase 2 completed if the peer (Cisco/Juniper) initiates phase 1
> rekeying meanwhile?

Yes, I guess, technically the CHILD_SA is rekeyed after the new SA is established. RFC 7296, section 2.8 states:

```
To rekey a Child SA within an existing IKE SA, create a new,
equivalent SA (see Section 2.17 below), and when the new one is
established, delete the old one.
```

However, the example in section 1.3.2. does not explicitly show the exchange to delete the old CHILD_SA as part of the rekeying process.

> If we increase the difference between phase 1 and phase 2 lifetimes
> the system becomes more stable, the tunnels stay up as collision will not
> happen so often.

You could also reduce the lifetimes on the strongSwan box to force it to be the initiator of rekeyings.

> We don't know but our guess is "no, Cisco would not likely delete them explicitly".

Which would probably not be correct according to the quote above (since all CHILD_SAs are adopted by the rekeyed IKE_SA the old CHILD_SA should still get deleted). But delaying the deletion probably is.

> However, would it be worth of trying to make strongSwan IKEv2 to behave the same way as IKEv1? What do you think?

It would probably require quite some work to change the state to CHILD_REKEYED and handle this properly in all cases (which could lead to subtle issues I guess). But I haven't really looked into this in detail.

> What comes to the INVALID_SPI message, we are seeing it during every successful and unsuccessful phase 2 rekeying.
> Our conclusion is that somehow we manage to send one data packet with an invalid SPI which is
> not anymore/yet valid from the Cisco point of view. This seems not to be any problem as it happens
> really for every phase 2 rekeying.

The Cisco box apparently sends this if it receives ESP packets using the new SPI while, as initiator of the rekeying, it has not yet been able to install the new CHILD_SA (also see #1291). This can be seen in the Cisco log of the successful case:

```
*Jan 31 18:28:32.948: %CRYPTO-4-RECVD_PKT_INV_SPI: decaps: rec'd IPSEC packet has invalid spi for destaddr=1.1
.31.9,...
*Jan 31 18:28:32.952: IKEv2:(SA ID = 2):Sending INVALID_SPI notify
```

It delays that message due to the window size and then sends it after it received the CREATE_CHILD_SA response.

> So far we have no answer to this question.

I guess you could try changing the notify at source:src/libcharon/sa/ikev2/tasks/ike_rekey.c#L225 from NO_PROPOSAL_CHOSEN to TEMPORARY_FAILURE to test what happens.

#4 - 16.02.2016 10:03 - Lasse Huovinen

Sorry for the delayed reply, we did some further experimenting and took a while.

```
I guess you could try changing the notify at source:src/libcharon/sa/ikev2/tasks/ike_rekey.c#L225 from NO_PROP
OSAL_CHOSEN to TEMPORARY_FAILURE to test what happens.
```

We changed libcharon as per your suggestion to send TEMPORARY_FAILURE instead of NO_PROPOSAL_CHOSEN in the case of collision. Unfortunately the both devices seem to just ignore the message.

```
But even if we'd e.g. switch the state to CHILD_REKEYED (we use that for IKEv1 only, currently) I wonder if th
e Cisco box would delete the old SA after rekeying the IKE_SA or if it would just remain installed until it ex
pires (or the IKE_SA is terminated).

> However, would it be worth of trying to make strongSwan IKEv2 to behave the same way as IKEv1? What do you t
hink?

>> It would probably require quite some work to change the state to CHILD_REKEYED and handle this properly in
all cases (which could lead to subtle issues I guess). But I haven't really looked into this in detail.
```

Do you happen to have any plans to change strongSwan's behavior to handle the collision with IKEv2? We would be happy to test the changes.

If you have no plans and if we tried to do this change by ourselves, would you be willing to give us some pointers where to start and what should we consider to avoid breaking the software?
What would be your estimate how big task the change would be, are we talking about hours, days, weeks?

#5 - 16.02.2016 10:29 - Tobias Brunner

> We changed libcharon as per your suggestion to send TEMPORARY_FAILURE instead of NO_PROPOSAL_CHOSEN in the case of collision. Unfortunately the both devices seem to just ignore the message.

What does "ignore" mean? Do they retry rekeying later or do they consider the rekeying failed? Or do they really just ignore the message and wait for another response to the CREATE_CHILD_SA exchange (which would be very strange)?

> Do you happen to have any plans to change strongSwan's behavior to handle the collision with IKEv2? We would be happy to test the changes.

Probably, not sure when though.

> If you have no plans and if we tried to do this change by ourselves, would you be willing to give us some pointers where to start and what should we consider to avoid breaking the software?

In the ikev2 tasks (source:src/libcharon/sa/ikev2/tasks). During rekeying and in particular in case of collisions several of them work together. The ways to break the whole thing are very many, so I don't really have any recommendations.

> What would be your estimate how big task the change would be, are we talking about hours, days, weeks?

I guess a few days. But some of the more subtle issues might only be revealed later during use (i.e. not by us, as we don't do much testing beyond the automated tests, which currently don't cover rekeying or collisions).

#6 - 18.02.2016 10:24 - Lasse Huovinen

Here are the log snippets showing how Cisco and Juniper behave upon reception of the "TEMP_FAIL" message. Neither one of the devices support it.

Cisco does not recognize "TEMP_FAIL" message and it deletes the SA as it considers rekeying failed.
Afterwards strongSwan establishes IKE SA and probably because we have "closeaction=restart".

```
strongSwan
==========
Feb 17 04:39:26 04[NET] waiting for data on sockets
Feb 17 04:39:26 16[NET] <signal_policy|4401> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (428 byt
es)
Feb 17 04:39:26 16[ENC] <signal_policy|4401> parsed CREATE_CHILD_SA request 6 [ SA No KE N(SET_WINSIZE) ]
Feb 17 04:39:26 16[IKE] <signal_policy|4401> peer initiated rekeying, but a child is half-open
Feb 17 04:39:26 16[ENC] <signal_policy|4401> generating CREATE_CHILD_SA response 6 [ N(TEMP_FAIL) ]
```

```
Feb 17 04:39:26 16[NET] <signal_policy|4401> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76 bytes
)
Feb 17 04:39:26 05[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb 17 04:39:26 04[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
Feb 17 04:39:26 04[NET] waiting for data on sockets
Feb 17 04:39:26 13[NET] <signal_policy|4401> received packet: from 1.1.31.9[500] to 10.104.31.13[500] (76 byte
s)
Feb 17 04:39:26 13[ENC] <signal_policy|4401> parsed INFORMATIONAL request 7 [ ]
Feb 17 04:39:26 13[ENC] <signal_policy|4401> generating INFORMATIONAL response 7 [ ]
Feb 17 04:39:26 13[NET] <signal_policy|4401> sending packet: from 10.104.31.13[500] to 1.1.31.9[500] (76 bytes
)
Feb 17 04:39:26 05[NET] sending packet: from 10.104.31.13[500] to 1.1.31.9[500]
Feb 17 04:39:26 04[NET] received packet: from 1.1.31.9[500] to 10.104.31.13[500]
```

```
Cisco
=====
Feb 17 02:39:26.367: IKEv2:(SA ID = 4):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : D16E3BD4F7EF8FCE - Responder SPI : 44781E77574636BB Message id: 6
IKEv2 CREATE_CHILD_SA Exchange RESPONSE
Payload contents:
 NOTIFY(Unknown - 43)

Feb 17 02:39:26.367: IKEv2:(SA ID = 4):Processing any notify-messages in child SA exchange
Feb 17 02:39:26.367: IKEv2:(SA ID = 4):Validating create child message
Feb 17 02:39:26.367: IKEv2:(SA ID = 4):An expected payload is missing from the packet

Feb 17 02:39:26.367: IKEv2:(SA ID = 4):
Feb 17 02:39:26.367: IKEv2:(SA ID = 4):Create child exchange failed

Feb 17 02:39:26.367: IKEv2:(SA ID = 4):
Feb 17 02:39:26.367: IKEv2:(SA ID = 4):IKE SA rekey failed
Feb 17 02:39:26.367: IKEv2:(SA ID = 4):Abort exchange
Feb 17 02:39:26.367: IKEv2:(SA ID = 6):Deleting SA

Feb 17 02:39:26.371: IKEv2:(SA ID = 4):Sending Packet [To 10.104.31.13:500/From 1.1.31.9:500/VRF i0:f0]
Initiator SPI : D16E3BD4F7EF8FCE - Responder SPI : 44781E77574636BB Message id: 7
IKEv2 INFORMATIONAL Exchange REQUEST
Payload contents:
 ENCR

Feb 17 02:39:26.383: IKEv2:(SA ID = 4):Received Packet [From 10.104.31.13:500/To 1.1.31.9:500/VRF i0:f0]
Initiator SPI : D16E3BD4F7EF8FCE - Responder SPI : 44781E77574636BB Message id: 7
IKEv2 INFORMATIONAL Exchange RESPONSE
Payload contents:
```

Juniper does not recognize TEMP_FAIL either. Instead of explicitly deleting the old SA looks like it establishes a new one. strongSwan detects the duplicate.

```
strongSwan
==========
Feb  5 10:28:35 09[NET] <signal_policy|25> received packet: from 10.106.49.18[500] to 10.104.49.2[500] (476 by
tes)
Feb  5 10:28:35 09[ENC] <signal_policy|25> parsed CREATE_CHILD_SA request 11 [ SA No KE N(SET_WINSIZE) V ]
Feb  5 10:28:35 09[IKE] <signal_policy|25> peer initiated rekeying, but a child is half-open
Feb  5 10:28:35 09[ENC] <signal_policy|25> generating CREATE_CHILD_SA response 11 [ N(TEMP_FAIL) ]
Feb  5 10:28:35 09[NET] <signal_policy|25> sending packet: from 10.104.49.2[500] to 10.106.49.18[500] (76 byte
s)
Feb  5 10:28:35 06[NET] sending packet: from 10.104.49.2[500] to 10.106.49.18[500]
Feb  5 10:28:35 03[CFG] proposing traffic selectors for us:
Feb  5 10:28:35 03[CFG]  10.106.49.2/32
Feb  5 10:28:35 03[CFG] proposing traffic selectors for other:
Feb  5 10:28:35 03[CFG]  0.0.0.0/0[tcp/ssh]
Feb  5 10:28:35 03[CFG] proposing traffic selectors for us:
Feb  5 10:28:35 03[CFG]  10.106.49.2/32[tcp/ssh]
Feb  5 10:28:35 03[CFG] proposing traffic selectors for other:
Feb  5 10:28:35 03[CFG]  0.0.0.0/0
Feb  5 10:28:35 05[NET] received packet: from 10.106.49.18[500] to 10.104.49.2[500]
Feb  5 10:28:35 05[NET] waiting for data on sockets
Feb  5 10:28:35 11[NET] <27> received packet: from 10.106.49.18[500] to 10.104.49.2[500] (464 bytes)
Feb  5 10:28:35 11[ENC] <27> parsed IKE_SA_INIT request 0 [ SA KE No N(NATD_S_IP) N(NATD_D_IP) V ]
Feb  5 10:28:35 11[CFG] <27> looking for an ike config for 10.104.49.2...10.106.49.18
Feb  5 10:28:35 11[CFG] <27>   candidate: 10.104.49.2...10.106.49.18, prio 3100
Feb  5 10:28:35 11[CFG] <27> found matching ike config: 10.104.49.2...10.106.49.18 with prio 3100
Feb  5 10:28:35 11[ENC] <27> received unknown vendor ID: 69:93:69:22:87:41:c6:d4:ca:09:4c:93:e2:42:c9:de:19:e7
:b7:c6:00:00:00:05:00:00:05:00
```

```
Feb  5 10:28:35 11[IKE] <27> 10.106.49.18 is initiating an IKE_SA
Feb  5 10:28:35 11[IKE] <27> IKE_SA (unnamed)[27] state change: CREATED => CONNECTING
Feb  5 10:28:35 11[CFG] <27> selecting proposal:
Feb  5 10:28:35 11[CFG] <27>   proposal matches
Feb  5 10:28:35 11[CFG] <27> received proposals: IKE:AES_CBC_256/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_2048
Feb  5 10:28:35 11[CFG] <27> configured proposals: IKE:AES_CBC_256/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_2048
Feb  5 10:28:35 11[CFG] <27> selected proposal: IKE:AES_CBC_256/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_2048
Feb  5 10:28:35 11[ENC] <27> generating IKE_SA_INIT response 0 [ SA KE No N(NATD_S_IP) N(NATD_D_IP) ]
Feb  5 10:28:35 11[NET] <27> sending packet: from 10.104.49.2[500] to 10.106.49.18[500] (432 bytes)
Feb  5 10:28:35 06[NET] sending packet: from 10.104.49.2[500] to 10.106.49.18[500]
Feb  5 10:28:35 05[NET] received packet: from 10.106.49.18[500] to 10.104.49.2[500]
Feb  5 10:28:35 05[NET] waiting for data on sockets
Feb  5 10:28:35 16[NET] <27> received packet: from 10.106.49.18[500] to 10.104.49.2[500] (236 bytes)
Feb  5 10:28:35 16[ENC] <27> parsed IKE_AUTH request 1 [ IDi IDr AUTH SA TSi TSr N(INIT_CONTACT) N(SET_WINSIZE
) ]
Feb  5 10:28:35 16[CFG] <27> looking for peer configs matching 10.104.49.2[10.104.49.2]...10.106.49.18[10.106.
49.18]
Feb  5 10:28:35 16[CFG] <27>   candidate "signal_policy", match: 20/20/3100 (me/other/ike)
Feb  5 10:28:35 16[CFG] <signal_policy|27> selected peer config 'signal_policy'
Feb  5 10:28:35 16[IKE] <signal_policy|27> authentication of '10.106.49.18' with pre-shared key successful
Feb  5 10:28:35 16[IKE] <signal_policy|25> destroying duplicate IKE_SA for peer '10.106.49.18', received INITI
AL_CONTACT
Feb  5 10:28:35 16[IKE] <signal_policy|25> IKE_SA signal_policy[25] state change: ESTABLISHED => DESTROYING
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.1.1.104/32 === 10.102.49.16/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.1.1.104/32 === 10.102.49.16/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.102.49.16/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.102.49.16/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.1.1.104/32 === 10.102.49.16/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting SAD entry with SPI cb82680f  (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleted SAD entry with SPI cb82680f (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting SAD entry with SPI b4371041  (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleted SAD entry with SPI b4371041 (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.1.1.104/32 === 10.102.49.32/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.1.1.104/32 === 10.102.49.32/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.32/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.102.49.32/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.32/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.102.49.32/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.1.1.104/32 === 10.102.49.32/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.32/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.32/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting SAD entry with SPI cd37eabc  (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleted SAD entry with SPI cd37eabc (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting SAD entry with SPI 445e1995  (mark 0/0x00000000)
```

```
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleted SAD entry with SPI 445e1995 (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.1.1.104/32 === 10.102.49.16/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.1.1.104/32 === 10.102.49.16/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.102.49.16/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> policy still used by another CHILD_SA, not removed
Feb  5 10:28:35 16[KNL] <signal_policy|25> updating policy 10.102.49.16/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.1.1.104/32 === 10.102.49.16/29 out  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 in  (mark 0/0x000
00000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting policy 10.102.49.16/29 === 10.1.1.104/32 fwd  (mark 0/0x00
000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting SAD entry with SPI c43fa8ed  (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleted SAD entry with SPI c43fa8ed (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleting SAD entry with SPI 5dc02c90  (mark 0/0x00000000)
Feb  5 10:28:35 16[KNL] <signal_policy|25> deleted SAD entry with SPI 5dc02c90 (mark 0/0x00000000)
Feb  5 10:28:35 16[IKE] <signal_policy|27> authentication of '10.104.49.2' (myself) with pre-shared key
Feb  5 10:28:35 16[IKE] <signal_policy|27> successfully created shared key MAC
Feb  5 10:28:35 16[IKE] <signal_policy|27> IKE_SA signal_policy[27] established between 10.104.49.2[10.104.49.
2]...10.106.49.18[10.106.49.18]
Feb  5 10:28:35 16[IKE] <signal_policy|27> IKE_SA signal_policy[27] state change: CONNECTING => ESTABLISHED
```

```
Juniper
=======
[Feb  5 15:26:12]ikev2_packet_allocate: Allocated packet e85c00 from freelist
[Feb  5 15:26:13]ikev2_packet_allocate: Allocated packet ea8800 from freelist
[Feb  5 15:26:13]iked_pm_ike_spd_notify_received: Received authenticated notification payload unknown from loc
al:10.106.49.18 remote:10.104.49.2 IKEv2 for P1 SA 812730
[Feb  5 15:26:13]ikev2_decode_packet: [ea8800/14bc800] Received packet: HDR, unknown
[Feb  5 15:26:13]ikev2_state_ike_rekey_initiator_in: [ea8800/14bc800] Error: Mandatory payloads (SAr,Ni) missi
ng or extra payloads
[Feb  5 15:26:13]ikev2_process_notify: [ea8800/14bc800] Received error notify unknown (43)
[Feb  5 15:26:13]ikev2_state_error: [ea8800/14bc800] Negotiation failed because of error unknown (43)
[Feb  5 15:26:13]IKE negotiation fail for local:10.106.49.18, remote:10.104.49.2 IKEv2 with status: unknown
[Feb  5 15:26:13]IKE SA delete called for p1 sa 812732 (ref cnt 1) local:10.106.49.18, remote:10.104.49.2, IKE
v2
[Feb  5 15:26:13]Freeing all P2 SAs for IKEv2 p1 SA 812732
[Feb  5 15:26:13]iked_pm_p1_sa_destroy:  p1 sa 812732 (ref cnt 0), waiting_for_del 0x0
[Feb  5 15:26:13]IKE SA delete called for p1 sa 812730 (ref cnt 1) local:10.106.49.18, remote:10.104.49.2, IKE
v2
[Feb  5 15:26:13]Freeing all P2 SAs for IKEv2 p1 SA 812730
```

Thank you Tobias for your help. Now we at least understand what's causing the breaks we are seeing occasionally, and have better understanding what options we have to cope with the problem. One potential option is to try fixing the collision as you suggested. Lets see if and when we would have competence and time to do it.

Cheers, Lasse

#7 - 31.05.2016 21:16 - Kris Jobs

Hi, Lasse. I have this issue out there too, do you have any solution for it yet? Thanks :)

Lasse Huovinen wrote:

> Here are the log snippets showing how Cisco and Juniper behave upon reception of the "TEMP_FAIL" message. Neither one of the devices support it.
>
> Cisco does not recognize "TEMP_FAIL" message and it deletes the SA as it considers rekeying failed.
> Afterwards strongSwan establishes IKE SA and probably because we have "closeaction=restart".
>
> [...]
>
> Juniper does not recognize TEMP_FAIL either. Instead of explicitly deleting the old SA looks
> like it establishes a new one. strongSwan detects the duplicate.

[...]

Thank you Tobias for your help. Now we at least understand what's causing the breaks we are seeing occasionally, and have better understanding what options we have to cope with the problem. One potential option is to try fixing the collision as you suggested. Lets see if and when we would have competence and time to do it.

Cheers, Lasse

#8 - 03.06.2016 09:02 - Lasse Huovinen

Hi Kris,

So far we have been using long lifetimes and that seems to prevent collisions.
But a proper fix might available soon.

```
Lasse
```

#9 - 06.06.2016 09:49 - Tobias Brunner

- Tracker changed from Issue to Bug

- Status changed from Feedback to Assigned

- Assignee set to Tobias Brunner

- Target version set to 5.5.0

#10 - 04.07.2016 15:39 - Tobias Brunner

- Category set to libcharon

- Status changed from Assigned to Closed

- Resolution set to Fixed