strongSwan - Bug #1537

IKEv1: Deleting IKE-SA during QUICK_MODE when Phase 1 is ESTABLISHED will never delete the IKE-SA

27.06.2016 21:13 - prateek shankar

Status:	Closed	**Start date:**	27.06.2016
Priority:	Normal	**Due date:**	
Assignee:	Tobias Brunner	**Estimated time:**	0.00 hour
Category:	libcharon		
Target version:	5.5.1		
Affected version:	5.4.0	**Resolution:**	Fixed

Description

Hi,

I am able to reproduce this issue with IKEv1 between strongswan and a non-strongswan client both being initiators. Below are the detailed steps to reproduce this issue, (Although I could not reproduce the exact same issue between 2 strongswan)
Note: Am using swanctl to operate strongswan here.

1. Initially when strongswan is up and client is down, Strongswan initiates the connection. IKEv1 Phase 1 is in 'connecting' mode. Phase 1 tasks are active and QUICK_MODE is queued.

```
6fd252fb-815e-41f5-bdd8-940cc0d9ea81: #4, CONNECTING, IKEv1, a80bf9bda13c3322_i* 0000000000000000_
r
  local  '%any' @ 10.15.1.254[500]
  remote '%any' @ 51.1.1.1[500]
  queued:  QUICK_MODE QUICK_MODE
  active:  ISAKMP_VENDOR ISAKMP_CERT_PRE MAIN_MODE ISAKMP_CERT_POST ISAKMP_NATD
```

2. When client is up with, strongswan loads an IKEv1 connection and tries to initiate QUICK_MODE. IKEv1 Phase 1 gets established but QUICK_MODE is queued both in 'active' and 'queued' list.

```
ab88e862-81b8-484c-aaa4-969f719223cd: #4, ESTABLISHED, IKEv1, ed1d94aed05caa9e:a51c076b630526d6
  local  '50.1.1.1' @ 10.15.1.254[500]
  remote '51.1.1.1' @ 51.1.1.1[500]
  AES_CBC-192/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_2048
  established 12s ago, rekeying in 79094s
  queued:  QUICK_MODE
  active:  QUICK_MODE
```

3. After this point, if I issue a IKE-SA terminate ISAKMP_DELETE action is queued in 'queued' list.

```
6fd252fb-815e-41f5-bdd8-940cc0d9ea81: #4, ESTABLISHED, IKEv1, a80bf9bda13c3322_i* 5fe68352fb84b8ec
_r
  local  '50.1.1.1' @ 10.15.1.254[500]
  remote '51.1.1.1' @ 51.1.1.1[500]
  AES_CBC-192/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_2048
  established 73s ago, rekeying in 79469s
  queued:  QUICK_MODE ISAKMP_DELETE QUICK_MODE
  active:  QUICK_MODE
```

4. After this point, ISAKMP_DELETE task never gets executed. Because since QUICK_MODE tries out 5 retransmissions, gets reset. ISAKMP_DELETE is erased from both the queues and QUICK_MODE is queued again. Have pasted charon logs when this happens.

```
6fd252fb-815e-41f5-bdd8-940cc0d9ea81: #10, ESTABLISHED, IKEv1, a1fc5d7bb6086405_i* a7c004948a04221
2_r
  local  '50.1.1.1' @ 10.15.1.254\[500\]
```

```
  remote '51.1.1.1' @ 51.1.1.1\[500\]
  AES_CBC-192/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_2048
  established 1s ago, rekeying in 80044s
  queued:  QUICK_MODE QUICK_MODE
  active:  QUICK_MODE

Jun 27 19:03:21 14[CFG] vici terminate IKE_SA '6fd252fb-815e-41f5-bdd8-940cc0d9ea81'
Jun 27 19:03:27 04[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> sending retransmit 4 of request
message ID 3465307493, seq 4
Jun 27 19:03:27 04[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> sending packet: from 10.15.1.254
[500] to 50.1.1.1[500] (172 bytes)
Jun 27 19:03:29 13[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> sending retransmit 4 of request
message ID 3741647669, seq 4
Jun 27 19:03:29 13[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> sending packet: from 10.15.1.254
[500] to 51.1.1.1[500] (172 bytes)
Jun 27 19:04:09 11[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> sending retransmit 5 of request
message ID 3465307493, seq 4
Jun 27 19:04:09 11[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> sending packet: from 10.15.1.254
[500] to 50.1.1.1[500] (172 bytes)
Jun 27 19:04:11 04[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> sending retransmit 5 of request
message ID 3741647669, seq 4
Jun 27 19:04:11 04[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> sending packet: from 10.15.1.254
[500] to 51.1.1.1[500] (172 bytes)
Jun 27 19:05:24 09[KNL] creating delete job for CHILD_SA ESP/0xce395042/10.15.1.254
Jun 27 19:05:24 09[JOB] CHILD_SA ESP/0xce395042/10.15.1.254 not found for delete
Jun 27 19:05:24 05[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> giving up after 5 retransmits
Jun 27 19:05:24 05[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> initiating Main Mode IKE_SA dbd9
e93b-7288-4910-9e3d-dabc763a6f88[15] to 50.1.1.1
Jun 27 19:05:24 05[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> generating ID_PROT request 0 [ S
A V V V V ]
Jun 27 19:05:24 05[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|13> sending packet: from 10.15.1.254
[500] to 50.1.1.1[500] (160 bytes)
Jun 27 19:05:24 04[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> received packet: from 50.1.1.1[5
00] to 10.15.1.254[500] (128 bytes)
Jun 27 19:05:24 04[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> parsed ID_PROT response 0 [ SA V
 V ]
Jun 27 19:05:24 04[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> received DPD vendor ID
Jun 27 19:05:24 04[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> received unknown vendor ID: ba:c
7:c1:d1:9a:44:d6:af:51:af:a2:e9:60:96:98:4f
Jun 27 19:05:24 04[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> generating ID_PROT request 0 [ K
E No ]
Jun 27 19:05:24 04[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending packet: from 10.15.1.254
[500] to 50.1.1.1[500] (324 bytes)
Jun 27 19:05:24 08[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> received packet: from 50.1.1.1[5
00] to 10.15.1.254[500] (308 bytes)
Jun 27 19:05:24 08[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> parsed ID_PROT response 0 [ KE N
o ]
Jun 27 19:05:24 08[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> generating ID_PROT request 0 [ I
D HASH ]
Jun 27 19:05:24 08[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending packet: from 10.15.1.254
[500] to 50.1.1.1[500] (76 bytes)
Jun 27 19:05:24 06[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> received packet: from 50.1.1.1[5
00] to 10.15.1.254[500] (76 bytes)
Jun 27 19:05:24 06[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> parsed ID_PROT response 0 [ ID H
ASH ]
Jun 27 19:05:24 06[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> IKE_SA dbd9e93b-7288-4910-9e3d-d
abc763a6f88[15] established between 10.15.1.254[50.1.1.1]...50.1.1.1[50.1.1.1]
Jun 27 19:05:24 06[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> scheduling rekeying in 81813s
Jun 27 19:05:24 06[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> maximum IKE_SA lifetime 90453s
Jun 27 19:05:24 06[ENC] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> generating QUICK_MODE request 29
04747605 [ HASH SA No ID ID ]
Jun 27 19:05:24 06[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending packet: from 10.15.1.254
[500] to 50.1.1.1[500] (172 bytes)
Jun 27 19:05:27 09[KNL] creating delete job for CHILD_SA ESP/0xcd07ea41/10.15.1.254
Jun 27 19:05:27 09[JOB] CHILD_SA ESP/0xcd07ea41/10.15.1.254 not found for delete
Jun 27 19:05:27 16[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> giving up after 5 retransmits
Jun 27 19:05:27 14[CFG] vici connection 47 unknown
```

```
Jun 27 19:05:27 16[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> initiating Main Mode IKE_SA 6fd252fb-815e-41f5-bdd8-940cc0d9ea81[16] to 51.1.1.1
Jun 27 19:05:27 16[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> generating ID_PROT request 0 [ SA V V V V ]
Jun 27 19:05:27 16[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|14> sending packet: from 10.15.1.254[500] to 51.1.1.1[500] (160 bytes)
Jun 27 19:05:27 07[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> received packet: from 51.1.1.1[500] to 10.15.1.254[500] (128 bytes)
Jun 27 19:05:27 07[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> parsed ID_PROT response 0 [ SA V V ]
Jun 27 19:05:27 07[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> received DPD vendor ID
Jun 27 19:05:27 07[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> received unknown vendor ID: ba:c7:c1:d1:9a:44:d6:af:51:af:a2:e9:60:96:98:4f
Jun 27 19:05:27 07[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> generating ID_PROT request 0 [ KE No ]
Jun 27 19:05:27 07[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending packet: from 10.15.1.254[500] to 51.1.1.1[500] (324 bytes)
Jun 27 19:05:27 04[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> received packet: from 51.1.1.1[500] to 10.15.1.254[500] (308 bytes)
Jun 27 19:05:27 04[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> parsed ID_PROT response 0 [ KE No ]
Jun 27 19:05:27 04[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> generating ID_PROT request 0 [ ID HASH ]
Jun 27 19:05:27 04[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending packet: from 10.15.1.254[500] to 51.1.1.1[500] (76 bytes)
Jun 27 19:05:27 09[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> received packet: from 51.1.1.1[500] to 10.15.1.254[500] (76 bytes)
Jun 27 19:05:27 09[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> parsed ID_PROT response 0 [ ID HASH ]
Jun 27 19:05:27 09[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> IKE_SA 6fd252fb-815e-41f5-bdd8-940cc0d9ea81[16] established between 10.15.1.254[50.1.1.1]...51.1.1.1[51.1.1.1]
Jun 27 19:05:27 09[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> scheduling rekeying in 81975s
Jun 27 19:05:27 09[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> maximum IKE_SA lifetime 90615s
Jun 27 19:05:27 09[ENC] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> generating QUICK_MODE request 1906900257 [ HASH SA No ID ID ]
Jun 27 19:05:27 09[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending packet: from 10.15.1.254[500] to 51.1.1.1[500] (172 bytes)
Jun 27 19:05:28 04[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending retransmit 1 of request message ID 2904747605, seq 4
Jun 27 19:05:28 04[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending packet: from 10.15.1.254[500] to 50.1.1.1[500] (172 bytes)
Jun 27 19:05:31 14[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending retransmit 1 of request message ID 1906900257, seq 4
Jun 27 19:05:31 14[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending packet: from 10.15.1.254[500] to 51.1.1.1[500] (172 bytes)
Jun 27 19:05:35 05[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending retransmit 2 of request message ID 2904747605, seq 4
Jun 27 19:05:35 05[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending packet: from 10.15.1.254[500] to 50.1.1.1[500] (172 bytes)
Jun 27 19:05:38 10[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending retransmit 2 of request message ID 1906900257, seq 4
Jun 27 19:05:38 10[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending packet: from 10.15.1.254[500] to 51.1.1.1[500] (172 bytes)
Jun 27 19:05:48 13[IKE] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending retransmit 3 of request message ID 2904747605, seq 4
Jun 27 19:05:48 13[NET] <dbd9e93b-7288-4910-9e3d-dabc763a6f88|15> sending packet: from 10.15.1.254[500] to 50.1.1.1[500] (172 bytes)
Jun 27 19:05:51 14[IKE] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending retransmit 3 of request message ID 1906900257, seq 4
Jun 27 19:05:51 14[NET] <6fd252fb-815e-41f5-bdd8-940cc0d9ea81|16> sending packet: from 10.15.1.254[500] to 51.1.1.1[500] (172 bytes)
```

I found 2 issues created and fixed earlier related to this: https://wiki.strongswan.org/issues/429 and https://wiki.strongswan.org/issues/1410
Also, if i try out the patch provided in #1410 with few additions like in this PR https://github.com/strongswan/strongswan/pull/48
I was able to see this issue fixed.
Wanted to ask is there a reason why the patch wasn't merged to master.

Related issues:		
Related to Feature #1410: Deleting an IKEv1 IKE_SA can't take effect immediat...	**Closed**	**15.04.2016**
Related to Issue #429: ipsec down is not bringing down a connection!	**Closed**	

Associated revisions

Revision 60d0f52f - 19.07.2016 11:48 - Tobias Brunner

ike1: Flush active queue when queueing a delete of the IKE_SA

By aborting the active task we don't have to wait for potential
retransmits if the other peer does not respond to the current task.
Since IKEv1 has no sequential message IDs and INFORMATIONALs are no real
exchanges this should not be a problem.

Fixes #1537
References #429, #1410
Closes strongswan/strongswan#48

History

#1 - 28.06.2016 12:29 - Tobias Brunner

- Description updated

- Status changed from New to Feedback

- Assignee set to Tobias Brunner

> 2. When client is up with, strongswan loads an IKEv1 connection and tries to initiate QUICK_MODE. IKEv1 Phase 1 gets established but QUICK_MODE is queued both in 'active' and 'queued' list.

What's the reason the Quick Mode exchange does not get completed?

> 3. After this point, if I issue a IKE-SA terminate ISAKMP_DELETE action is queued in 'queued' list.

Why is there also yet another QUICK_MODE task in the queue?

> 4. After this point, ISAKMP_DELETE task never gets executed.

Yes, that's discussed in #1410.

> Wanted to ask is there a reason why the patch wasn't merged to master.

Which patch? The one in #1410 is merged. The one in #429 won't work for IKEv2, so that will never get merged in that form. The one in the PR I don't like due to the version check (and I don't think flushing the queued tasks is really necessary as the ISAKMP_DELETE task is preferred over others with the patch in #1410). But as I mentioned in #1410-3 it is probably possible to cancel the active task for IKEv1 before queueing the delete as INFORMATIONAL messages are no exchanges and message IDs are not sequential.

I pushed an IKEv1-specific version of the patch in #429 to the *1537-ikev1-force-delete* branch.

#2 - 28.06.2016 12:30 - Tobias Brunner

- Related to Feature #1410: Deleting an IKEv1 IKE_SA can't take effect immediately when there are other tasks in the task queue added

#3 - 28.06.2016 12:30 - Tobias Brunner

- Related to Issue #429: ipsec down is not bringing down a connection! added

#4 - 19.07.2016 02:53 - prateek shankar

Hi Tobias, The branch 1537-ikev1-force-delete you gave actually solves this problem. Would you be merging this to the master.?

#5 - 19.07.2016 11:52 - Tobias Brunner

- Tracker changed from Issue to Bug

- Target version set to 5.5.1

- Resolution set to Fixed

> The branch 1537-ikev1-force-delete you gave actually solves this problem. Would you be merging this to the master.?

I just did.

#6 - 19.07.2016 11:52 - Tobias Brunner

- Status changed from Feedback to Closed

#7 - 20.07.2016 03:54 - prateek shankar

Thanks Tobias,
Also just wanted to ask. Is there a way to check if the latest commit you made has passed all automated test cases.? Or do you merge only after the tests pass.?

#8 - 20.07.2016 11:18 - Tobias Brunner

> Is there a way to check if the latest commit you made has passed all automated test cases.? Or do you merge only after the tests pass.?

If you are referring to unit tests check the Travis CI results. Running the integration and regression test scenarios is currently not automated, we run them before releases (also developers releases and release candidates) and sometimes when developing and before merging specific changes. I did not in this particular instance as it's the only commit since 5.5.0 and there are no side-effects that would manifest themselves in this environment (deleting the IKE_SA is pretty much the last thing that happens in the scenarios).