strongSwan - Bug #556

TLS Close notify incompatible with Windows 7/8 Agile Client (EAP-TTLS/EAP-PEAP)

29.03.2014 17:05 - Anthony Tom

Status:	Closed	**Start date:**	29.03.2014
Priority:	Normal	**Due date:**	
Assignee:	Martin Willi	**Estimated time:**	0.00 hour
Category:	interoperability		
Target version:	5.3.0		
Affected version:	5.1.2	**Resolution:**	Fixed

Description

When using Windows Agile Clients (Windows 7 and Windows 8.1 tested, 64bit arch) as an initiator and with EAP-PEAP or EAP-TTLS client authentication, the TLS Close notify packet sent by the StrongSwan (in responder mode) causes an immediate Windows Agile Client disconnect after a successful authentication if EAP-PEAP is used, or a connection timeout if EAP-TTLS is used. The TLS close notify packet is not understood by the Agile Client. To restore compatibility, commit 7bbf7aa97a0acf3d728f should be reverted. Attached are two Charon's logs:

StrongSwan-5.1.2-vanilla.log - showcases the behavior of the vannila Strongswan 5.1.2
StrongSwan-5.1.2-reverted-commit.log - showcases the behavior of the vannila Strongswan 5.1.2 with revert of the problematic patch

The most notable difference is that the vanilla StrongSwan 5.1.2 tries to send TLS close notify and in that way continues the EAP conversation, while the StrongSwan 5.1.2 compiled after the the reversal of the commit sends (in this case - authentication is successful) EAP/SUCC:

Vanilla:

Mar 28 22:54:01 dev-1 charon: 02[IKE] received tunneled EAP-TTLS AVP [EAP/RES/MSCHAPV2]
Mar 28 22:54:01 dev-1 charon: 02[IKE] EAP_TTLS phase2 authentication of 'iketest' with EAP_MSCHAPV2 successful
Mar 28 22:54:01 dev-1 charon: 02[TLS] sending TLS close notify
Mar 28 22:54:01 dev-1 charon: 02[ENC] generating IKE_AUTH response 11 [EAP/REQ/TTLS]

Windows Agile client believes the EAP conversation is almost over (expects EAP/SUCC or failure), so this "IKE_AUTH response 11" packet seems strange to it. It stalls (EAP-TTLS) or disconnects (EAP-PEAP).

Reverted commit:

Mar 29 15:55:11 dev-1 charon: 09[IKE] received tunneled EAP-TTLS AVP [EAP/RES/MSCHAPV2]
Mar 29 15:55:11 dev-1 charon: 09[IKE] EAP_TTLS phase2 authentication of 'iketest' with EAP_MSCHAPV2 successful
Mar 29 15:55:11 dev-1 charon: 09[IKE] EAP method EAP_TTLS succeeded, MSK established
Mar 29 15:55:11 dev-1 charon: 09[ENC] generating IKE_AUTH response 8 [EAP/SUCC]

Windows Agile Client expects EAP/SUCC packet or a failure and does recieve, in this case, EAP/SUCC. StrongSwan then goes on setting up routes and with the rest of the connection setup process.

This issue is easy to reproduce and reverting the mentioned commit resolves it. In my opinion this commit creates a big compatibility issue and that's why I think it is urgent to fix it in the next stable release. Also, this issue is present, probably since the commit was done, and that's a year ago.

BTW. thank you for the wonderful software (StrongSwan) and all your efforts you've put into it :)

Associated revisions

Revision e62ff799 - 05.02.2015 09:20 - Martin Willi

libtls: Don't send TLS close notifies in EAP after application succeeds

With the introduction of PT-TLS, we started sending TLS close notifies after
the application layer completes. While this makes sense for TCP based
transports, it is not required in EAP methods. In EAP, handshake completion
can be directly signaled using the outer EAP-SUCCESS message. This also saves
one round-trip in the EAP exchange.

Windows 7/8 does not seem to like TLS close notifies at all in EAP, and either stalls (EAP-TTLS) or disconnects (PEAP).

Fixes #556.

Revision 970378c5 - 19.02.2015 11:29 - Martin Willi

libtls: Don't send TLS close notifies in EAP after application succeeds

With the introduction of PT-TLS, we started sending TLS close notifies after the application layer completes (7bbf7aa9). While this makes sense for TCP based transports, it is not required in EAP methods. In EAP, handshake completion can be directly signaled using the outer EAP-SUCCESS message. This also saves one round-trip in the EAP exchange.

Windows 7/8 does not seem to like TLS close notifies at all in EAP, and either stalls (EAP-TTLS) or disconnects (PEAP).

Fixes #556.

History

#1 - 14.04.2014 12:51 - Tobias Brunner

- Description updated

- Priority changed from Urgent to High

- Target version deleted (5.1.3)

#2 - 14.04.2014 12:51 - Tobias Brunner

- Priority changed from High to Normal

#3 - 05.02.2015 02:07 - Hrvoje Maracic

This bug still affects the client on the latest OS version (tried today on Windows Server 2012 R2 with all updates) and the latest CentOS 7 strongSwan package (5.2.0). Could someone please look into it?

#4 - 05.02.2015 09:24 - Martin Willi

- Status changed from New to Feedback

- Assignee set to Martin Willi

Hi,

Please try the referenced commit, it restores the old behavior by not sending TLS close notifies in the context of EAP methods.

Regards
Martin

#5 - 19.02.2015 10:57 - Wouter Smeltekop

I too was affected by this exact issue. I was trying to get Windows 8.1 to connect using PEAP-MSChapV2 but ended up pulling my hair out on why everything seemed to work according to the logs until StrongSwan closed the TLS channel.

I couldn't make the GIT PPA work for some reason so I ended up using this guide on compiling StrongSwan for Ubuntu:

[[http://danielpocock.com/using-debcheckout-to-build-strongswan-5.0-on-debian-wheezy]]

For some reason it did not have this commit included. I used nano to manually edit the files involved in the commit and compiled everything. I can confirm that with this commit my Windows 8.1 machine is able to connect now using PEAP-MSChapV2.

THANK YOU

#6 - 19.02.2015 11:32 - Martin Willi

- Status changed from Feedback to Closed

- Target version set to 5.3.0

- Resolution set to Fixed

Thanks for the feedback. I've merged the fix to master, closing the issue.

Files

StrongSwan-5.1.2-vanilla.log	8.42 KB	29.03.2014	Anthony Tom
ipsec-vanilla.conf	447 Bytes	29.03.2014	Anthony Tom
StrongSwan-5.1.2-reverted-commit.log	7.71 KB	29.03.2014	Anthony Tom
ipsec-reverted-commit.conf	447 Bytes	29.03.2014	Anthony Tom