strongSwan - Feature #1353

Support for 96-bit HMAC-SHA-256 using IANA integrity algorithm #12

16.03.2016 04:26 - David Bartley

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Tobias Brunner	**Estimated time:**	0.00 hour
Category:	libcharon		
Target version:	5.5.3		
Resolution:	Fixed		

Description

I'm trying to connect to a 3rd-party that is uses the 96-bit variant of HMAC-SHA-256. Unfortunately, they're unable to support any other integrity algorithm. At least Juniper hardware (what the 3rd-party is using) and racoon (the software that I'm trying to migrate away from) use the 96-bit variant, and doing a bit of research also indicates Cisco uses the 96-bit variant, though I haven't confirmed that.

Related issues:

Related to Issue #2324: Specifying esp and ike for Linux 2.6.27	**Closed**

Associated revisions

Revision 4366494d - 26.05.2017 11:23 - Tobias Brunner

Merge branch 'sha-256-96'

Adds an option to locally configure 96-bit truncation for HMAC-SHA256
when negotiated using the official algorithm identifier. This is for
compatibility with peers that incorrectly use this shorter truncation
(like Linux does by default).

Fixes #1353.

History

#1 - 16.03.2016 09:39 - Tobias Brunner

- Status changed from New to Feedback

I hope Cisco does not use 96-bit truncation. And the others are just wrong. The Linux kernel did this incorrectly too at some point, but that was fixed with 2.6.33. strongSwan may negotiate the use of SHA-256 with 96-bit truncation (via *sha256_96* see IKEv2CipherSuites) but it uses an identifier from the private use range. So this only works for strongSwan-strongSwan communication between an old and a new kernel. But I guess you could patch strongSwan so the actually sent identifier is not 1026 but 12 instead.

#2 - 16.03.2016 09:42 - David Bartley

Tobias Brunner wrote:

> I hope Cisco does not use 96-bit truncation. And the others are just wrong. The Linux kernel did this incorrectly too at some point, but that was fixed with 2.6.33. strongSwan may negotiate the use of SHA-256 with 96-bit truncation (via *sha256_96* see IKEv2CipherSuites) but it uses an identifier from the private use range. So this only works for strongSwan-strongSwan communication between an old and a new kernel. But I guess you could patch strongSwan so the actually sent identifier is not 1026 but 12 instead.

I could do that, but I'd rather not maintain a patched copy of strongswan; it would be nice if there was a new config option or string (e.g. "sha256_96_old") that could be used.

#3 - 23.03.2016 11:23 - Tobias Brunner

> I could do that, but I'd rather not maintain a patched copy of strongswan; it would be nice if there was a new config option or string (e.g. "sha256_96_old") that could be used.

Adding a config option to send 12 as identifier if *sha256_96* is configured wouldn't be that much of a problem. But what do we do when we receive 12 as integrity algorithm identifier. We have no way of knowing what truncation the peer will actually use, so we'd have to use some heuristic (like checking the local proposal for *sha256_96* and patching the received proposal so it will match, which is currently not possible).

If you can't get the other vendors to fix their implementations it might really be easier if you just patch your setup and map the identifiers back and forth (source:src/libcharon/encoding/payloads/proposal_substructure.c#L1442, source:src/libcharon/encoding/payloads/proposal_substructure.c#L552) or simply change the truncation to 96-bit when the SA is added to the kernel (source:src/libcharon/plugins/kernel_netlink/kernel_netlink_ipsec.c#L1383).

#4 - 23.03.2016 11:52 - David Bartley

Tobias Brunner wrote:

> Adding a config option to send 12 as identifier if *sha256_96* is configured wouldn't be that much of a problem. But what do we do when we receive 12 as integrity algorithm identifier. We have no way of knowing what truncation the peer will actually use, so we'd have to use some heuristic (like checking the local proposal for *sha256_96* and patching the received proposal so it will match, which is currently not possible).
>
> If you can't get the other vendors to fix their implementations it might really be easier if you just patch your setup and map the identifiers back and forth (source:src/libcharon/encoding/payloads/proposal_substructure.c#L1442, source:src/libcharon/encoding/payloads/proposal_substructure.c#L552) or simply change the truncation to 96-bit when the SA is added to the kernel (source:src/libcharon/plugins/kernel_netlink/kernel_netlink_ipsec.c#L1383).

I did end up patching kernel_netlink_ipsec.c (simply swapping the sha256_96/sha256 IDs didn't work). In my case I do know that the other peer will always truncate to 96-bits, so having a "truncated = yes" option in the "conn" section that enabled/disabled truncation in netlink would work. If you don't know what truncation the other party supports, it might be easier to just use sha1 instead of sha256.

#5 - 28.03.2017 11:38 - Markus Sattler

- File 0110-sha2-truncation.patch added

We had the needed to force SHA2 96Bit truncation on connection based,
find attached a patch to make this possible.
default is 128Bit until sha2_96_truncate=yes is added to the connection.

#6 - 09.05.2017 17:35 - Noel Kuntze

If you want this to be applied upstream, you need to add the necessary license headers to your files.

#7 - 09.05.2017 17:48 - Tobias Brunner

> If you want this to be applied upstream, you need to add the necessary license headers to your files.

I'm not sure if that's necessary as these are rather trivial changes. But there are lots of whitespace and other code style issues.

#8 - 09.05.2017 21:04 - Noel Kuntze

I think because it adds a configuration option and touches many files, it is complex enough to require a license.

#9 - 10.05.2017 08:42 - Markus Sattler

- File 0110-sha2-truncation.patch added

Feel free to integrate the patch upstream, sure will help the community with some compatibility problems with SHA2 out there.
Find attached a version with Copyright and license information.

#10 - 10.05.2017 15:49 - Tobias Brunner

> I think because it adds a configuration option and touches many files, it is complex enough to require a license.

No, I wouldn't want copyright headers just for one-liners, and really the changes are absolutely trivial.

#11 - 11.05.2017 08:14 - Tobias Brunner

- Tracker changed from Issue to Feature

- Target version set to 5.5.3

#12 - 15.05.2017 17:42 - Tobias Brunner

- Related to Issue #2324: Specifying esp and ike for Linux 2.6.27 added

#13 - 26.05.2017 11:26 - Tobias Brunner

- Category set to libcharon

- Status changed from Feedback to Closed

- Assignee set to Tobias Brunner

- Resolution set to Fixed

Files

0110-sha2-truncation.patch	5.72 KB	28.03.2017	Markus Sattler
0110-sha2-truncation.patch	7.16 KB	10.05.2017	Markus Sattler