strongSwan - Issue #73

StrongSwan IKEv2 CERT Payload missing first 0x30 byte of DER-encoding.

30.03.2009 19:07 - Martin Willi

Status:	Closed		
Priority:	High		
Assignee:	Martin Willi		
Category:	charon		
Affected version:		**Resolution:**	

Description

StrongSwan's IKEv1 CERT Payload appears correct as per the RFC and inter-operates correctly with other implementations. The CERT Payload, per the RFC, for CertEncoding 4, (x.509 Certificate - Signature) is to be DER-encoded. This means the first byte (big-endian) should be 0x30 - ASN.1 SEQUENCE-OF. In IKEv1 (pluto) it is. ..

In IKEv2 (charon), however, the first byte is chopped off, leaving the remaining DER-encoded cert payload un-readable by strict asn.1 parsers.

Here's the first few bytes of the OpenSSL-generated DER-encoded certificate (little endian):

0000000 - 8230 0305 8230 eb02 03a0 0102 0202 0501
0000010 - 0d30 0906 862a 8648 0df7 0101 0505 3000

In big-endian the first few bytes are 0x3082 0503
That's exactly what we get back from pluto for the CERT payload.
However, for IKEv2, we get this:

Certificate Payload Next: 0x00 (None) Flags: 0x00 () Len: 0x0508 (1288)
CERT Cert Encoding 4 (X.509_Certificate_-_Signature)
8204FF30 8202E7A0 03020102 02010630 |...0.....0|
0D06092A 864886F7 0D010105 05003016 |...*.H.....|
31143012 06035504 03140B49 4E544552 |1.0...U....INTER|

Notice the first 0x30 byte is missing.

History

#1 - 31.03.2009 09:30 - Martin Willi

- Status changed from New to Closed

- Affected version set to invalid

The data field of a received CERT payload looks perfectly fine here:

```
11[ENC]    0: 30 82 04 0D 30 82 02 F5 A0 03 02 01 02 02 01 03  0...0...........
...
```

We also never experienced interoperability problems with other vendors, I don't think we are encoding something wrong here.

However, your hex dumps seem to be screwed up, you're dumping little endian integers, but the data is plain binary. Please revise your hex dump function...

#2 - 06.05.2013 21:33 - Andreas Steffen

- Tracker changed from Bug to Issue