strongSwan - Bug #9

Order of the Nonce and KE payload in IKE_SA_INIT must conform to RFC 4306

13.07.2007 16:29 - Andreas Steffen

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Andreas Steffen	**Estimated time:**	0.00 hour
Category:	charon		
Target version:	4.1.5		
Affected version:	5.9.0	**Resolution:**	

Description

In order to conform with RFC 4306 the KE payload must be sent before the Nonce payload:

```
HDR, SAi1, KEi, Ni   -->

<--    HDR, SAr1, KEr, Nr, [CERTREQ]
```

The current order in sa/tasks/ike_init.c:build_payloads() is:

```
message->add_payload(message, (payload_t*)sa_payload);

nonce_payload = nonce_payload_create();
    nonce_payload->set_nonce(nonce_payload, this->my_nonce);
    message->add_payload(message, (payload_t*)nonce_payload);

ke_payload = ke_payload_create_from_diffie_hellman(this->dh);
    message->add_payload(message, (payload_t*)ke_payload);
```

History

#1 - 16.07.2007 09:07 - Martin Willi

- Status changed from New to Closed

- Affected version set to fixed

Yes we do this the wrong way around. But the code is used twice, for initial IKE_SA_INIT setup, but also for rekeying in CREATE_CHILD_SA message. For rekeying, the order is the other way round (RFC4306 1.3):

```
       HDR, SK {[N], SA, Ni, [KEi],
           [TSi, TSr]}             -->
```

I think I've fixed that once for rekeying, but didn't realized that IKE_SA_INIT uses another payload order (which is kinda strange, IMHO).

Fixed in r2965.

#2 - 16.07.2007 10:06 - Martin Willi

> I think I've fixed that once for rekeying, but didn't realized that IKE_SA_INIT uses another payload order (which is kinda strange, IMHO).

Yes I've "fixed" that for rekeying, see r2423.