**TO:** T10 Membership
**FROM:** Paul A. Suhler, Quantum Corporation
David Black, EMC
**DATE:** 22 October 2008
**SUBJECT:** T10/08-346r1, SPC-4: Correction to IKEv2-SCSI Certificate Request Payload

## 1 Revisions

0 Initial revision (28 August 2008)

1 First revision (22 October 2008)
Fixed title of Table 408.
Reword description of CERTIFICATION AUTHORITY field and modify table.
Require device server to be able to store at least four certification authority values.
Explicitly permit multiple certificate request payloads.

## 2 General

The definition of the Certificate Request payload in IKEv2-SCSI (SPC-4 clause 7.6.3.5.5) does not match that in IKEv2 (RFC 4306 clause 3.7). This proposal describes the problem, suggests alternatives, and proposes a fix based on one alternative.

Red highlighting in this section is used for emphasis, not to indicate changes.

### *2.1 The Mismatch*

In IKEv2, the Certificate Encoding field is used differently between the Certificate and Certificate Request payloads, and the following field is named and used differently (Certificate Data and Certification Authority, respectively). However, in IKEv2-SCSI a single format is used for both payloads and the CERTIFICATE ENCODING and CERTIFICATE DATA fields are defined identically for those payloads.

RFC 4306 clause 3.7 (certificate request payload) says in the first bullet after table 13:

> o Certificate Encoding (1 octet) - Contains an encoding of the type or format of certificate requested. Values are listed in section 3.6.

And two paragraphs later:

> The Certificate Encoding field has the same values as those defined in section 3.6. The Certification Authority field contains an indicator of trusted authorities for this certificate type. The Certification Authority value is a concatenated list of SHA-1 hashes of the public keys of trusted Certification Authorities (CAs). Each is encoded as the SHA-1 hash of the Subject Public Key Info element (see section 4.1.2.7 of [RFC3280]) from each Trust Anchor certificate. The twenty-octet hashes are concatenated and included with no other formatting.

In other words, the Certificate Encoding field does not describe the contents of the adjacent Certification Authority field. This is different from the Certificate Encoding field in the Certificate payload; in that case, the Certificate Encoding field ***does*** describe the contents of the Certification Authority field.

However, IKEv2-SCSI makes no distinction between the use of the fields in the Certificate Request and Certificate payloads:

> The CERTIFICATE ENCODING field describes the contents of the CERTIFICATE DATA field and shall contain one of the values shown in table 409.

The results of this mismatch are:

1) the node sending the Certificate Request payload has no way to explicitly specify the desired format for the requested certificate; and
2) the trusted authorities list is not in a standard format.

The authors of the original IKEv2-SCSI proposal have recommended fixing the problem:

"Be like IKE."
- Matt Ball

"I like IKEv2."
- Ralph Weber

David Black did not stoop to a pun, but was supportive.

### *2.2 Alternatives*

There seem to be three ways to fix this problem. One is to change the description of the Certificate Request payload in SPC-4 to correspond to RFC 4306. The disadvantage of this is that there may already exist implementations compliant with SPC-4r16.

The second (suggested by David Black) would be to define a new Certificate Request payload with CERTIFICATE ENCODING and CERTIFICATE DATA fields that conform to IKEv2 and deprecate the existing Certificate Request payload. This might require assigning a different value for the NEXT PAYLOAD field, which would not match the current value of 26h, which is consistent with that defined in IKEv2. This would still leave a mismatch.

The third would be to leave the contents of the Certificate Request payload as they are currently defined, but to state that:

1) the contents of the CERTIFICATE DATA field are one or more root CA certificates of trust anchors; and
2) the requested certificate should also be in the format specified by the CERTIFICATE ENCODING field.

This would result in a CERTIFICATE DATA field about two orders of magnitude longer than that in alternative one or two.

The remainder of this proposal suggests changes to SPC-4 to implement the first alternative.

### *2.3 Trust Anchors*

A new item in revision r1 is to require that a device server support at least four trust anchors. This requirement is borrowed from FC-SP. The intent is that one would be for the device vendor, one for the customer, and a spare slot for each to allow changes. The method for modifying the trust anchors is beyond the scope of either SPC-4 or RFC 4306.

## 3 Proposed changes to SPC-4r16

The proposed changes separate the descriptions of the Certificate and Certificate Request payloads into separate subclauses. As is done in RFC 4306, the description of the Certificate payload precedes the description of the Certificate Request payload. The final field in the Certificate Request payload is renamed to match that in RFC 4306.

There are several cross-references that are updated, including two that were initially to the wrong subclause.

Unless otherwise indicated, additions are shown in blue, deletions in ~~red strikethrough~~, and comments in green.

Change to 5.14.4.2, fifth paragraph following Figure 12:

The optional Certificate Request payload or payloads (see ~~7.6.3.5.5~~7.6.3.5.6) enables the device server to request a certificate from the application client. If the Authentication step is being skipped (see 5.14.4.1), the device server shall not include ~~any~~ a Certificate Request ~~payloads~~ payload in the parameter data. Use of the Certificate Request payload is described in 5.14.4.3.

Change to 5.14.4.2, fifth paragraph following Figure 13:

The optional Certificate Request payload or payloads (see ~~7.6.3.5.5~~7.6.3.5.6) allows an application client to request the delivery of a Certificate payload (see 7.6.3.5.5) in the parameter data for the Authentication step SECURITY PROTOCOL IN command (see 5.14.4.3).

Change to 5.14.4.8.3, first numbered list:

5) Zero or more Certificate Request payloads (see ~~5.14.4.3~~7.6.3.5.6).

Change to 5.14.4.9.2, last numbered list on page:

4) Zero or more Certificate Request payloads (see ~~5.14.4.3~~7.6.3.5.6);

Change to Table 401, Reference column for Certificate Request: ~~7.6.3.5.5~~7.6.3.5.6

### 5.14.4.3 Handling of the Certificate Request payload and the Certificate payload

As detailed in this subclause, a Certificate Request payload (see ~~7.6.3.5.5~~7.6.3.5.6) in one set of parameter data requests the delivery of a Certificate payload (see 7.6.3.5.5) in the next set of parameter data transferred. The purpose of these IKEv2-SCSI protocol elements is as follows:

<< no additional changes in this subclause >>

### 7.6.3.5.5 Certificate payload ~~and Certificate Request payload~~

~~The Certificate Request payload (see table 408) allows an application client or device server to request the use of certificates as part of identity authentication and to name one or more trust anchors (see RFC 4306) for the certificate verification process.~~ The Certificate payload (see table 408) delivers a requested identity authentication certificate. The protocol for using ~~Certificate Request payloads and~~ Certificate payloads is described in 5.14.4.3.

**Table 408 – ~~Certificate Request payload and~~ Certificate payload format**

| Byte \ Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | NEXT PAYLOAD | | | | | | | |
| 1 | CRIT (1b) | Reserved | | | | | | |
| 2 | (MSB) | IKE PAYLOAD LENGTH (n+1) | | | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | CERTIFICATE ENCODING | | | | | | | |
| 5 | | CERTIFICATE DATA | | | | | | |
| n | | | | | | | | |

The NEXT PAYLOAD field, CRIT bit, and IKE PAYLOAD LENGTH field are described in 7.6.3.5.1.

The CRIT bit is set to one in the ~~Certificate Request payload and~~ the Certificate payload.

The CERTIFICATE ENCODING field describes the contents of the CERTIFICATE DATA field and shall contain one of the values shown in table 409.

**Table 409 – CERTIFICATE ENCODING field**

| Code | Description | Reference |
|---|---|---|
| 00h | Reserved | |
| 01h to 03h | Prohibited | Annex C |
| 04h | X.509 Certificate – Signature | RFC 4306 |
| 05h to 0Ah | Prohibited | Annex C |
| 0Bh | Raw RSA Key | RFC 4306 and RFC 4718 |
| 0Ch to 0Dh | Prohibited | Annex C |
| 0Eh to C8h | Restricted | IANA |
| C9h to FFh | Reserved | |

The contents of the CERTIFICATE DATA field depend on the value in the CERTIFICATE ENCODING field.

The relationship between the Certificate payload and the Identification payload is described in 7.6.3.5.4.

Device servers that support certificates should support a mechanism outside the scope of this standard for replacing certificates and have the ability to store more than one certificate to facilitate such replacements.

### 7.6.3.5.6 Certificate Request payload

The Certificate Request payload (see table x) allows an application client or device server to request the use of certificates as part of identity authentication and to name one or more trust anchors (see RFC 4306) for the certificate verification process. The protocol for using Certificate Request payloads is described in 5.14.4.3.

**Table x –Certificate Request payload format**

| Bit / Byte | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | NEXT PAYLOAD | | | | | | | |
| 1 | CRIT (1b) | Reserved | | | | | | |
| 2 | (MSB) | IKE PAYLOAD LENGTH (n+1) | | | | | | |
| 3 | | | | | | | | (LSB) |
| 4 | CERTIFICATE ENCODING | | | | | | | |
| Certification authority list | | | | | | | | |
| 5 | (MSB) | CERTIFICATION AUTHORITY [first] | | | | | | |
| 24 | | | | | | | | (LSB) |
| | . . . | | | | | | | |
| n-19 | (MSB) | CERTIFICATION AUTHORITY [last] | | | | | | |
| n | | | | | | | | (LSB) |

The NEXT PAYLOAD field, CRIT bit, and IKE PAYLOAD LENGTH field are described in 7.6.3.5.1.

The CRIT bit is set to one in the Certificate Request payload.

The value in the CERTIFICATE ENCODING field indicates the type or format of certificate requested and shall contain one of the values shown in table 409. Multiple Certificate Request payloads may be included in the parameter data transferred by a single command, and should have different values of the CERTIFICATE ENCODING field.

Each CERTIFICATION AUTHORITY contains an indicator of a trusted authority for this certificate type. The indicator is a SHA-1 hash of the public key of a trusted Certification Authority (CA). The indicator is encoded as the SHA-1 hash of the Subject Public Key Info element from the Trust Anchor certificate (see RFC 3280).

Device servers that support certificates should support a mechanism outside the scope of this standard for replacing certification authority values and shall have the ability to store at least four certification authority values to facilitate such replacements.

Existing subclauses 7.6.3.5.6 through 7.6.3.5.14 are renumbered to 7.6.3.5.7 through 7.6.3.5.15.

Change to Annex C, lettered list item o):

o) The usage of Certificate Encodings in the Certificate payload and Certificate Request payload (see ~~7.6.3.5.5~~7.6.3.5.6) are constrained as follows:

<< No further changes are specified in this proposal. >>

Editor: Please check all references to:

a) Certificate payloads;
b) Certificate Request payloads; and
c) Relevant sections of model clause.