EMC²
where information lives®

To: INCITS Technical Committee T10
From: David L. Black, EMC
Email: black_david@emc.com
Date: October 29, 2008
Subject: SPC-4: Digital Signature Authentication (08-423r0)

1) *Revision history*

Revision 0 (Oct 29, 2008) First revision (r0)

2) *Related documents*

spc4r16 – SCSI Primary Commands – 4
IETF RFC 4306 – Internet Key Exchange (IKEv2) Protocol
IETF RFC 5280 – Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile
IETF RFC 5282 – Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol

3) *Overview*

Review of the IKEv2-SCSI specification has identified text that is incorrect when digital signatures are used for authentication. The erroneous text specifies that the recipient calculates the expected authentication value. This is applicable when a shared secret key is used for authentication (Shared Key Message Integrity Code). This text is not correct when a digital signature is used, as a digital signature is computed with the sender's private key and the recipient checks it with the sender's public key; the recipient can check the signature for correctness, but cannot reproduce the signature on its own.

This document proposes corrections to the specification of authentication computations to cover digital signature algorithms, including certificate validation. As part of these corrections, a table column is introduced to classify each authentication algorithm as Shared Key (ShKey) or Digital Signature (DSig).

This document also serves as a vehicle to make a couple of updates to Annex C, one of which has impacts elsewhere because it reflects IETF completion of standardization work motivated by IKEv2-SCSI.

Existing text is shown in **BLACK**, additions are shown in **GREEN**, text to be deleted is ~~**struck through in RED**~~ and comments (not to be included) are shown in **BLUE**.

Proposal:

2.5 IETF References

Add the following two references in the appropriate positions in the list of IETF RFCs:

RFC 5280, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*

RFC 5282, *Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol*

7.6.3.5.6 Authentication payload

In the middle of p.487 (after first numbered list):

When processing the Authentication step SECURITY PROTOCOL OUT command, the device server shall ~~compute the expected~~ check the contents of the AUTHENTICATION DATA field by applying the algorithm specified by the ALGORITHM IDENTIFIER field in the SA_AUTH_OUT IKEv2-SCSI cryptographic algorithm descriptor in the IKEv2-SCSI SA Cryptographic Algorithms payload (see 7.6.3.5.12) in the Key Exchange step (see 5.14.4.8) as described in table 431 (see 7.6.3.6.6) and this subclause to the following concatenation of bytes:

No change to list of bytes to concatenate. The following text insertion begins immediately after the list of bytes to concatenate:

The class of authentication algorithm, shared key or digital signature, is specified in table 431 (see 7.6.3.6.6).

For a shared key algorithm, the check shall consist of computing the expected contents of the AUTHENTICATION DATA field using the shared key associated with the identification data received in the Identification - Application Client payload (see 7.6.3.5.4) and comparing the expected contents to the actual contents of this field; the check is successful if the expected contents match the actual contents, otherwise the check is not successful.

For a digital signature algorithm, the check shall consist of using the public key associated with the identity determined from the identification data received in the Identification - Application Client payload (see 7.6.3.5.4) and any certificates received in the Certificate payload (see 7.6.3.5.5) to check the digital signature contents of the AUTHENTICATION DATA field; the check is successful when the signature is validated as a correct digital signature of the concatenation of bytes specified above, otherwise the check is not successful. If the public key used to validate the digital signature is obtained from a certificate, the certification path from the certificate to a trust anchor should be validated (see RFC 5280); the digital signature check is not successful if the path cannot be validated, independent of whether the digital signature is valid.

If the ~~expected~~ check of the contents of the AUTHENTICATION DATA field ~~do not match actual contents of the AUTHENTICATION DATA field,~~ is not successful, then:

In the middle of p.488 (after first numbered list):

After GOOD status is received for the Authentication step SECURITY PROTOCOL IN command, the application client should ~~compute the expected~~ check the contents of the AUTHENTICATION DATA field by applying the algorithm specified by the ALGORITHM IDENTIFIER field in the SA_AUTH_IN IKEv2-SCSI cryptographic algorithm descriptor in the IKEv2-SCSI SA Cryptographic Algorithms payload (see 7.6.3.5.12) in the Key Exchange step (see 5.14.4.8) as described in table 431 (see 7.6.3.6.6) and this subclause to the following concatenation of bytes:

No change to list of bytes to concatenate. The following text insertion begins immediately after the list of bytes to concatenate:

The class of authentication algorithm, shared key or digital signature, is specified in table 431 (see 7.6.3.6.6).

For a shared key algorithm, the check shall consist of computing the expected contents of the AUTHENTICATION DATA field using the shared key associated with the identification data received in the Identification - Device Server (see 7.6.3.5.4) payload and comparing the expected contents to the actual contents of this field; the check is successful if the expected contents match the actual contents, otherwise the check is not successful.

For a digital signature algorithm, the check shall consist of using the public key associated with the identity determined from the identification data in Identification - Device Server payload (see 7.6.3.5.4) and any certificates received in the Certificate payload (see 7.6.3.5.5) to check the digital signature contents of the AUTHENTICATION DATA field; the check is successful when the signature is validated as a correct digital signature of the concatenation of bytes specified above, otherwise the check is not successful. If the public key used to validate the digital signature is obtained from a certificate, the certification path from the certificate to a trust anchor should be validated (see RFC 5280); the digital signature check is not successful if the path cannot be validated, independent of whether the digital signature is valid.

If the ~~expected~~ check of the contents of the AUTHENTICATION DATA field ~~do not match actual contents of the AUTHENTICATION DATA field,~~ is not successful, then the application client should abandon the IKEv2-SCSI CCS and notify the device server that it is abandoning the IKEv2-SCSI CCS as described in 5.14.4.12.

7.6.3.6.2 Encryption algorithm (ENCR) IKEv2-SCSI cryptographic algorithm descriptor

Add "RFC 5282" as an additional Reference for the AES-CCM and AES-GCM rows in table 423.

7.6.3.6.6 IKEv2-SCSI authentication algorithm IKEv2-SCSI cryptographic algorithm descriptor

Add a new "Class" column to table 431 between the Description and Support columns (only these 3 columns are shown below):

Description	Class	Support
SA_AUTH_NONE	n/a	Optional
RSA Digital Signature with SHA-1 [a]	DSig	Optional
Shared Key Message Integrity Code	ShKey	Optional
ECDSA with SHA-256 on the P-256 curve [a]	DSig	Optional
ECDSA with SHA-512 on the P-521 curve [a]	DSig	Optional
Vendor Specific		Optional
Restricted		Prohibited
Reserved		

Add the following key for the Class column below the table entries, but above the footnotes:

Class Key: DSig - Digital Signature based on an asymmetric (public/private) key pair
ShKey - Shared Key based on a secret key

Add the following text immediately following table 431 (Q: Does this belong in the model clause somewhere?):

Note: Authentication algorithms provide proof of identity by demonstrating knowledge of a secret associated with the identity. A shared key authentication algorithm associates a secret key with the identity; that secret key is used to both generate and check the authentication data. A digital signature authentication algorithm is based on asymmetric cryptography; it associates a pair of keys with the identity, a signing key to generate digital signatures as authentication data and a verification key to check the signatures. The signing key is secret and is kept private to the entity that generates the digital signatures; it is called a private key. The verification key cannot be used to generate signatures on behalf of the associated identity and hence does not need to be kept secret; it is called a public key.

Note: Certificates and the IKEv2-SCSI Certificate payload provide a means of communicating public keys to IKEv2-SCSI entities that check digital signatures. A certificate is a data structure that contains an identity, a public key for that identity and additional relevant information that may constrain use of the public key. A certificate is signed by a certification authority (CA), and that digital signature is part of the certificate data structure.

Note: In the simplest case, an entity that trusts a CA uses the public key of the CA to verify the certificate; a successful verification establishes that the public key in the certificate is associated with the identity in the certificate. When a certificate is supplied as part of the IKEv2-SCSI authentication step, it allows a single CA public key to serve as a trust anchor for verification digital signatures from any identity that can obtain a certificate from the CA. This enables a single CA public key to be configured for authentication verification instead of configuring a public key for each identity that is to be authenticated. In full generality, multiple checks are involved in verifying a certificate and the verification may traverse a certification path composed of multiple certificates, see RFC 5280.

C.1 IKEv2 protocol details and variations for IKEv2-SCSI

Modify item y) and add a new item ab) at the end of the list:

y) ~~The description of how combined mode algorithms are used in the Encrypted payload in this standard predates the definition of equivalent functionality in IETF standards.~~ IETF standards omit the Integrity transform instead of using AUTH_COMBINED;

ab) The critical (CRIT) bit is set to 1b in all IKEv2-SCSI payloads specified in this standard because they are required to be recognized by all implementations of IKEv2-SCSI. IETF RFC 4306 sets the Critical (C) bit to zero in all IKEv2 payloads specified in RFC 4306, but requires that all IKEv2 implementations recognize all payloads specified in RFC 4306.