KMIP Trust Establishment Proposal

Subject: Proposal for using public key infrastructure (PKI) techniques for establishing trust between a KMIP client and a KMIP server.

Date: 6 October 2010

Contact: Judy Furlong, EMC/RSA

Contributors: Robert Griffin, EMC/RSA, David Lawson, Emulex, Larry Hofer, Emulex, Robert Nixon, Emulex

Revision History

- 17 September 2009: Initial version
- 9 July 2010: Updated Draft
- 30 September 2010: Draft for KMIP F2F
- 6 October 2010: Draft for KMIP TC

Overview

This proposal defines changes to KMIP v1 [1]. The proposal describes a method for how a KMIP client and a KMIP server exchange their respective public key certificates and establish a trust relationship with one another. Once established, the trust relationship enables implementation of additional layers of security controls on KMIP messages such as digitally signing the payload of a KMIP message.

References

[1] OASIS, *Key Management Interoperability Protocol (KMIP) Specification Version 1.0* Committee Specification 0.1, 15 June 2010.

[2] ISO/IEC 9798-3:1998 *Information technology -- Security techniques -- Entity authentication -- Part 3: Mechanisms using digital signature techniques*

1. Pre-Conditions

The trust establishment proposal assumes the following pre-conditions:

- The credential (either public key or public key certificate) for a Trusted Root Certification Authority (CA) has been transmitted out-of-bands and embedded in the KMIP client or otherwise pre-established.
- The KMIP client has generated or requested and loaded an asymmetric key pair and associated public key certificate. [Note: The KMIP client certificate is not expected to chain to the Trusted Root CA embedded in the KMIP client.]
- The KMIP server has created its own asymmetric key pair and has had its public key certified by a CA that chains to the Trusted Root CA whose credential is embedded in the KMIP client.

2. New KMIP Operations

This proposal defines two new KMIP operations, *Exchange Credential and Confirm Credential*, both of which are initiated by a KMIP client. These operations allow a KMIP client and a KMIP server to exchange and trust credentials such as public key certificates that will subsequently be used to authenticate and/or protect KMIP messages exchanged by these parties. This proposal assumes the exchanged credential is a public key certificate, but it's possible other types of credentials may be added in subsequent revisions.

2.1 Exchange Credential

This operation allows a KMIP client and KMIP server to exchange credentials such as public key certificates that will be used to authenticate and/or protect subsequent KMIP message exchanged by these parties.

The request which is generated by the KMIP client contains information about the type of credential being exchanged, a nonce value generated by the KMIP client and the credential of the KMIP client.

Request Payload		
Object	**REQUIRED**	**Description**
Object Type	Yes	Determines the type of credential object being exchanged. Initial version assumes X.509v3 public key certificate (or certificate chain).
IV/Counter/Nonce	Yes	Nonce value (N1) generated by KMIP client for this message.
Credential	Yes	The credential of the KMIP client. Initial version assumes X.509v3 certificate (or certificate chain).

Table 1: Exchange Credential Request Payload

The response which is generated by the KMIP server contains information about the type of credential being exchanged, the nonce value generated by the KMIP client and included in the request message, a nonce value generated by the KMIP server encrypted using the public key from the KMIP client certificate provided in the request message, and the credential of the KMIP server.

Response Payload		
Object	**REQUIRED**	**Description**
Object Type	Yes	Determines the type of credential object being exchanged. Initial version assumes X.509 public key certificate (or certificate chain).
IV/Counter/Nonce	Yes	Nonce value (N1) from the *Exchange Credential* request message.
Encrypted IV/Counter/Nonce	No	Nonce value (N2) generated by KMIP server and encrypted in public key extracted from the KMIP client

		certificate in the *Exchange Credential* request message. This encrypted nonce value must be included if the KMIP server requires that the KMIP client submit a *Confirm Credential* request.
Credential	Yes	The credential of the KMIP server. Initial version assumes X.509 certificate (or certificate chain).

Table 2: Exchange Credential Response Payload

Upon receipt of the *Exchange Credential* response, the KMIP client compares the nonce (N1) sent by the KMIP server to the one it generated and included in the *Exchange Credential* request. Assuming the two nonces match, the KMIP client will then verify the certificate (or certificate chain) of the KMIP server using the Trusted Root CA credential embedded in the KMIP client in the pre-condition, out-of-bands step. If verification of the KMIP server certificate is successful, the KMIP client stores the KMIP server certificate (or certificate chain) for future use. Finally, if the encrypted nonce (N2) is present in the *Exchange Credential* response, the KMIP client then uses its private key to decrypt the encrypted nonce (N2) which was generated by the KMIP server.

2.2 Confirm Credential

At the conclusion of a successful Exchange Credential operation, the KMIP client has authenticated the KMIP server by verifying its certificate (or certificate chain) and confirmed it has been communicating with a live KMIP server since the KMIP server returned the nonce generated by the KMIP client. However, the KMIP server has not confirmed that the client it has been communicating with is in fact the one which possesses the private key that corresponds to the KMIP client public key certificate.

In order to prove to the KMIP server that the KMIP client is in possession of the private key, the KMIP client can send a *Confirm Credential* request message. The request message contains a new nonce value that has been generated by the KMIP client, a decrypted version of the encrypted nonce received from the KMIP server in the *Exchange Credential* response, and the first nonce value generated by the KMIP client and provide in the *Exchange Credential* request.

Request Payload		
Object	**REQUIRED**	**Description**
IV/Counter/Nonce	Yes	New nonce value (N3) generated by KMIP client for this message.
IV/Counter/Nonce	Yes	Decrypted copy of the encrypted nonce value (N2) generated by KMIP server from the *Exchange Credential* response message
IV/Counter/Nonce	Yes	Nonce value (N1) generated by KMIP client obtained from the *Exchange Credential* request message.

Table 3: Confirm Credential Request Payload

Upon receipt of the *Confirm Credential* request, the KMIP server looks for the presence of the new nonce (N3) to ensure it's a fresh message and the initial nonce (N1) to tie it with the previous *Exchange Credential* request message. The server then compares unencrypted nonce (N2) with the value it generated and included in the *Exchange Credential* response message. If these nonces are the same,

the KMIP server has now confirmed that the KMIP client is in possession of the corresponding private key. The KMIP server can store the KMIP client certificate (or certificate chain) for future use.

Assuming all nonce checks are successful the KMIP server generates and sends a *Confirm Credential* response. The response message includes a copy of the new nonce (N3) from the *Confirm Credential* request.

Response Payload		
Object	***REQUIRED***	***Description***
IV/Counter/Nonce	Yes	New nonce value (N3) generated by KMIP client and obtained from the *Confirm Credential* request message.

Table 4: Confirm Credential Payload