Krishna Yellepeddy

23 March 2011

Draft proposal for supporting Key groups in KMIP

Use cases for key groups

1. Suggestion at the f2f was to have a group proposal for symmetric key groups only, the goal being to keep changes to the specification to a minimum

2. A symmetric key group is a collection of symmetric key group managed objects . All the keys in this group are of the same type and size, e.g. 128 bit AES keys, 256 bit AES keys etc.

3. Two models of devices /applications that use these symmetric key groups:
   a) Application/Device for which the server generates symmetric keys. The device requests the server for the next key in the group.
   b) Application/Device which registers symmetric keys with the key server in a specific symmetric key group and later requests the next key from this group.

4. The server support a locate next key operation on this key group. The server policy for determining the next key to serve from the group is outside the scope of the KMIP specification.

Server operations for supporting symmetric key groups

- Server infers the homogeneity criteria from the first key created or registered in the symmetric key group. For example:
  - Client creates a symmetric key group with a single 128 bit AES key. Client names this key group “aesSymmetricKeyGroup”.
  - Server infers that the key group is for 128 bit AES keys
  - If client then tries to add a symmetric key other than a 128 bit AES key in the key group “aesSymmetricKeyGroup”, server returns an error that this addition is not allowed.

- Server operation for getting next key in group
  - Server defines its own policy for identifying the next key in the group to server to any of the clients requesting keys from this group. How the server defines and implements this policy is outside the scope of the KMIP specification..

New KMIP attributes and operations needed for symmetric key group

New attributes:

- **Symmetric Key Group** – To assign a symmetric key object to a symmetric key group, the symmetric key group object SHOULD be set.

New operation:

- LocateNext - returns the uuid of the next key from the group.

  -- Locate Next operation requests that the server return the next Symmetric key object from a symmetric key group. The symmetric key group label is specified in the request. If it is not specified in the request, it is inferred by the server based on client identity and/or other criteria based on server policy. These criteria are outside the scope of the specification.

Symmetric Key Group Attribute

Object	Encoding	Required
Symmetric Key Group	Structure	
Label	Text String	No

Symmetric Key Group

In the event a symmetric key is put into a symmetric key group structure SHALL be created. The Label field is OPTIONAL

Symmetric Key Group Attribute

Symmetric Key Group

In the event a symmetric key is put into a symmetric key, a symmetric key group structure SHALL be created. The Label field is OPTIONAL

SHALL always have a value	No.
Initially set by	Client or Server
Modifiable by client	No
Modifiable by server	No
Deletable by client	No
Multiple instances permitted	No
When implicitly set	During Create or Register of a symmetric key
Applies to object types	Symmetric Key

Operations that are impacted

2 client to server operations need updates:

- Create/Register: When symmetric Key Group is specified as an attribute, server recognizes it is dealing with a symmetric key group and takes action accordingly.

In effect, with these additions we'll have support for "economy class" groups

Destroy needs to be considered by the sever implementers:

- Since destroy is an operation for which clients have very restricted access, server implementers should evaluate whether to support destroy of a symmetric key in a group.

Upgrade from economy class to first class

- In a future release, if we add Group as a managed object we can bridge from v1.1 as follows:
  - A future KMIP server with support for group as first class object will continue to support the LocateNext operation to be backward compatible with a v 1.1 client. So a v 1.1 client does not have to be changed for getting the next key from a group.
  - When a create or register of the first symmetric key is done for a symmetric key group by a v 1.1 client, this future server will create a first class Group object.
  - Existing symmetric key groups from a v 1.1 server are migrated into a first class group object.