KMIP Usage Guide Proposal on Templates

Created by: Indra Fitzgerald, HP
Version: 1.0
Date: 2/17/10
Purpose: Elaine Barker requested that the Usage Guide include a template example.

Section 3.6 of the KMIP Usage Guide will be replaced with the following:

3.6 Template

The usage of templates is an alternative approach for setting attributes in an operation request. Instead of individually specifying each attribute, a template may be used to set any of the following attributes for a managed object:

- Cryptographic Algorithm
- Cryptographic Length
- Cryptographic Domain Parameters
- Cryptographic Parameters
- Operation Policy Name
- Cryptographic Usage Mask
- Usage Limits
- Activation Date
- Process Start Date
- Protect Stop Date
- Deactivation Date
- Object Group
- Application Specific Information
- Contact Information
- Custom Attribute

In addition to these attributes, the template has attributes that are applicable to the template itself. These include the attributes (Unique Identifier, Initial Date, Last Change Date, and Archive Date) set implicitly after successfully completing a certain operation and attributes set by the client (Object Type and Name) in the Register request. When registering a template, the Name attribute for the template should be set. It is used to specify and identify the template in the Template-Attribute structure when attributes for a managed object are set.

The Template-Attribute structure allows for multiple template names and individual attributes to be specified in an operation request. The structure is used in the Create, Create Key Pair, Register, Re-key, Derive Key, Certify, and Re-certify operations. All of these operations with the exception of the Create Key Pair operation use the Template-Attribute tag. The Create Key Pair operation uses the Common Template-Attribute, Private Key Template Attribute, and Public Key Template-Attribute tags.

Templates may be the subject of the Register, Locate, Get, Get Attributes, Get Attribute List, Add Attribute, Modify Attribute, Delete Attribute, Delete Attribute, and Destroy operations. Clients are not able to create a template with the Create operation; instead templates are created using the Register operation. When the template is the subject of the operation, the Unique ID is used to identify the template. The template name is only used to identify the template inside a Template-Attribute structure.

3.6.1. Template Usage Examples

The purpose of these examples is to illustrate how templates are used. The first example shows how a template is registered. The second example shows how the newly registered template is used to create a symmetric key.

3.6.1 Registering a Template Example

In this example, a client registers a template by encapsulating attributes for creating a 256-bit AES key with the Cryptographic Usage Mask set to Encrypt and Decrypt.

The following is specified inside the Register Request Payload:

- Object Type: Template
- Template-Attribute:
  - Name: Template1
  - Cryptographic Algorithm: AES
  - Cryptographic Length: 256
  - Cryptographic Usage Mask: Encrypt and Decrypt
  - Operation Policy Name: OperationPolicy1

The Operation Policy OperationPolicy1 applies to the AES key being created using the template. It is not used to control operations on the template itself. KMIP does not allow operation policies to be specified for controlling operations on the template itself. The default policy for template objects is used for this purpose and is specified in the KMIP Specification.

3.6.2 Creating a Symmetric Key using a Template Example

In this example, the client uses the template created in example 3.6.1 to create a 256-bit AES key.

The following is specified in the Create Request Payload:

- Object Type: Symmetric Key
- Template-Attribute:
  - Name: Template1
  - Attribute:
    Name: AESkey
    Custom Attribute: x-ID74592

The Template-Attribute specifies both a template name and additional attributes. The Name attribute is not an attribute that may be set by a template. The Name attribute set for the template applies to the template itself (e.g., Template1 is the Name attribute of the Template object). The Name attribute for the symmetric key is therefore specified separately under Attribute. It is possible to specify the Custom Attribute inside the template; however, this particular example sets this attribute separately.