2.2.6 Template

A *Template* is a named Managed Object containing the client-settable attributes of a Managed Cryptographic Object (i.e., a stored, named list of attributes). A Template is used to specify the attributes of a new Managed Cryptographic Object in various operations. It is intended to be used to specify the cryptographic attributes of new objects in a standardized or convenient way. None of the client-settable attributes specified in a Template except the Name attribute apply to the template object itself, but instead apply to any object created using the Template.

The Template MAY be the subject of the Register, Locate, Get, Get Attributes, Get Attribute List, Add Attribute, Modify Attribute, Delete Attribute, and Destroy operations.

An attribute specified in a Template is applicable either to the Template itself or to objects created using the Template.

Attributes applicable to the Template itself are: Unique Identifier, Object Type, Name, Initial Date, Archive Date, and Last Change Date.

Attributes applicable to objects created using the Template are:

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

Object	Encoding	REQUIRED
Template	Structure	
Attribute	Attribute Object, see 2.1.1	Yes. MAY be repeated.

Table 133: Template Object Structure

4.11 Get

This operation requests that the server returns the Managed Object specified by its Unique Identifier.

Only a single object is returned. The response contains the Unique Identifier of the object, along with the object itself, which MAY be wrapped using a wrapping key as specified in the request.

If the object being returned is a Template, then it SHALL only contain Attributes mentioned in the list in section 2.2.6 as being applicable to objects created using the Template, and it SHALL contain the maximal known subset of said Attributes.

The following key format capabilities SHALL be assumed by the client restrictions apply when the client requests the server to return an object in a particular format:

- If a client registered a key in a given format, the server SHALL be able to return the key during the Get operation in the same format that was used when the key was registered.
- Any other format conversion MAY optionally be supported by the server.

Request Payload		
Object	**REQUIRED**	**Description**
Unique Identifier, see 3.1	No	Determines the object being requested. If omitted, then the ID Placeholder value is used by the server as the Unique Identifier.
Key Format Type, see 9.1.3.2.3	No	Determines the key format type to be returned.
Key Compression Type, see 9.1.3.2.2	No	Determines the compression method for elliptic curve public keys.
Key Wrapping Specification, see 2.1.6	No	Specifies keys and other information for wrapping the returned object. This field SHALL NOT be specified if the requested object is a Template.

Table 2152: Get Request Payload

Response Payload		
Object	**REQUIRED**	**Description**
Object Type, see 3.3	Yes	Type of object.
Unique Identifier, see 3.1	Yes	The Unique Identifier of the object.
Certificate, Symmetric Key, Private Key, Public Key, Split Key, Template, Secret Data, or Opaque Object, see 2.2	Yes	The cryptographic object being returned.

Table 3153: Get Response Payload