Key Management Interoperability Protocol (KMIP)

Contents

Key Management Interoperability Protocol (KMIP)

Key Management Interoperability Protocol (KMIP) is a client/server communication protocol for the storage and maintenance of key, certificate, and secret objects. The standard is governed by the Organization for the Advancement of Structured Information Standards (OASIS).

Information in this document helps you to understand the implementation of KMIP in IBM Security Key Lifecycle Manager.

Overview - Key Management Interoperability Protocol

The IBM Security Key Lifecycle Manager server supports Key Management Interoperability Protocol (KMIP) communication with clients for key management operations on cryptographic material. The material includes symmetric and asymmetric keys, certificates, and templates that are used to create and control their use.

The Key Management Interoperability Protocol is part of an Organization for the Advancement of Structured Information Standards (OASIS) standardization project for encryption of stored data and cryptographic key management.

For more information, see Key Management Interoperability Protocol documentation (http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=kmip).

KMIP attributes for keys and certificates

IBM Security Key Lifecycle Manager supports the following tasks:

- Managing following KMIP information through the IBM Security Key Lifecycle Manager graphical user interface:
  - Whether KMIP ports and timeout settings are configured.
  - Current KMIP certificate, indicating which certificate is in use for secure server or server/client communication.
  - Whether SSL/KMIP or SSL is specified for secure communication.
- You can update KMIP attributes for keys and certificates.

  For example, you can use the **tklmKeyAttributeUpdate** command to update:

  name
  Specifies the name that is used to identify or locate the object. This attribute is a Key Management Interoperability Protocol attribute.

  applicationSpecificInformation
  Specifies application namespace information as a Key Management Interoperability Protocol attribute.

  contactInformation
  Specifies contact information as a Key Management Interoperability Protocol attribute.

  cryptoParams ***cryptoparameter1, cryptoparameter2, ..., cryptoparameterN***
  Specifies the cryptographic parameters that are used for cryptographic

operations by using the object *cryptoparameter1, cryptoparameter2, ..., cryptoparameterN*. This attribute is a Key Management Interoperability Protocol attribute.

customAttribute
Specifies a custom attribute in string format as a Key Management Interoperability Protocol attribute. Client-specific attributes must start with the characters "x-" (x hyphen) and server-specific attributes must start with "y-" (y hyphen).

link
Specifies the link from one managed cryptographic object to another, closely related target managed cryptographic object. This attribute is a Key Management Interoperability Protocol attribute.

objectGroup
Specifies one or more object group names of which this object might be part. This attribute is a Key Management Interoperability Protocol attribute.

processStartDate
Specifies the date on which a symmetric key object can be used for process purposes. You cannot change the value after the date occurs. If you specify a date earlier than the current date, the value is set to the current date. This attribute is a Key Management Interoperability Protocol attribute.

protectStopDate
Specifies the date on which an object cannot be used for process purposes. You cannot change the value after the date occurs. If you specify a date earlier than the current date, the value is set to the current date. This attribute is a Key Management Interoperability Protocol attribute.

usageLimits
Specifies either total bytes (BYTE) or total objects (OBJECT) as a Key Management Interoperability Protocol attribute. You cannot modify this value once this object is used. For example, **GetUsageAllocation** calls this object.

- List and delete client-registered KMIP templates.

  Clients use a template to specify the cryptographic attributes of new objects in a standardized or convenient way. The template is a managed object that contains attributes in operations that the client can set for a cryptographic object. For example, the client can set application-specific information.

  tklmKMIPTemplateList
  List KMIP templates that IBM Security Key Lifecycle Manager provides. For example, you might list all templates.

  tklmKMIPTemplateDelete
  Delete KMIP templates that clients registered with IBM Security Key Lifecycle Manager.

- List and delete secret data such as passwords or a seed that is used to generate keys.

  tklmSecretDataDelete
  Delete secret data that KMIP clients sent to IBM Security Key Lifecycle Manager.

  tklmSecretDataList
  List secret data that KMIP clients sent to IBM Security Key Lifecycle Manager.

- Set default port and timeout properties

KMIPListener.ssl.port
Specifies the port on which the IBM Security Key Lifecycle Manager server listens for requests from libraries. The server communicates over the SSL socket by using Key Management Interoperability Protocol.

TransportListener.ssl.port
Specifies the port on which IBM Security Key Lifecycle Manager server listens for requests from tape libraries that communicate by using the SSL protocol.

TransportListener.ssl.timeout
Specifies how long the socket waits on a read() before closing. This property is used for the SSL socket.

- Enable or disable delete requests from KMIP clients.

  An authenticated client can request delete operations that might have a significant impact on the availability of a key, on server performance, and on key security. Specify the enableKMIPDelete attribute with either the **tklmDeviceGroupAttributeUpdate** or the **tklmDeviceGroupCreate** command to determine whether IBM Security Key Lifecycle Manager acts on these requests.

KMIP objects, attributes, and operations

KMIP elements include cryptographic objects, operations for the objects, and attributes that are associated with these objects. With the implementation of KMIP, you can manage cryptographic objects and control their use.

KMIP objects

KMIP supports the following cryptographic objects that are required by the client and the server for the key management operations.

Object	Description
Certificate	A digital certificate, such as an X.509 certificate.
Opaque Object	An object that is stored by a key management server, but not necessarily interpreted by it.
Private Key	The private portion of an asymmetric key pair.
Public Key	The public portion of an asymmetric key pair.
Secret Data	A shared secret value that is not a key or certificate.
Split Key	A secret, usually a symmetric key, or a private key, which is split into a number of parts, which can then be distributed to several key holders, for more security.
Symmetric Key	A symmetric encryption key or message authentication code key.
Template	A stored, named list of KMIP attributes.

KMIP operations

KMIP operations are the actions on the managed objects. For example, registering objects with the key management server, retrieving objects from the server, or destroying objects from the server.

Operation	Description
Activate	Requests that a managed object is to be activated. The request does not specify a Template object. The operation can be performed only on an object in the pre-active state. The operation changes the object state to Active, and sets the Activation Date to the current date and time.
AddAttribute	Adds an attribute instance or application-specific information instance to a managed object and sets its value.
Create	Generates a new symmetric key. You cannot use this operation to create a template but multiple templates can be included to simplify the key creation.
CreateKeyPair	Generates a new asymmetric key pair. You cannot use this operation to create a template, but multiple private key, public key, or common templates can be included to simplify the key pair creation.
DeleteAttribute	Deletes an attribute that is associated with a managed object. The object is specified by its unique identifier. Attributes are specified by their name.
Destroy	Destroys the key material for a managed object. Once the managed object is destroyed, its metadata is erased.
Get	Returns the managed object from the server, which is specified by its unique identifier.
GetAttributes	Requests for one or more attributes of a managed object. The object is specified by its unique identifier.
GetAttributeList	Requests a list of attribute names that are associated with the managed object. The object is specified by its unique identifier. This request supports application-specific information, custom attributes, and aliases.
Locate	Requests that the server search for one or more managed objects.
ModifyAttribute	Modifies the value of an existing attribute instance that is associated with a managed object. The object is specified by its unique identifier.
Query	Requests information about capabilities of the server and the protocol mechanisms.
Register	Registers a managed object that was created by the client or obtained by the client through some other means.
Revoke	Requests to revoke a managed cryptographic object or an opaque object.

KMIP attributes

KMIP attributes are properties of the objects. The attributes, which are associated with the objects, are named values stored by the key management server, and are obtained from the server via operations.

Attributes	Description
Application Specific Information	Stores data, which is specific to the applications by using the managed object.
Certificate Type	The type of a certificate, for example, X.509 or PGP. The Certificate Type value is set by the server when the certificate is created or registered.
Certificate Identifier	Provides the identification of a certificate.

Attributes	Description
Certificate Issuer	Identifies the issuer of a certificate, which contains the **Issuer Distinguished Name** from the **Issuer** field of the certificate.
Certificate Subject	Identifies the subject of a certificate.
Contact Information	Provides the user-defined contact information.
Cryptographic Algorithm	The algorithm that is used by the object, for example, DES or AES.
Cryptographic Length	The cleartext cryptographic key material length in bits.
Custom Attributes	Client or server defined attributes for vendor-specific purposes. The supported types are: Big Integer, Boolean, Byte String, Date-Time, Enumeration, Integer, Interval, Long Integer, and Text String.
Digest	Contains the digest value of the key or secret data, such as digest of the key material or digest of the certificate value.
Initial Date	Indicates the date and time when the managed object was first created or registered by the server.
Activation Date	Indicates the activation date and time of the managed cryptographic object.
Process Start Date	The date and time when a managed symmetric key object can begin to process cryptographically protected information. For example, decryption or unwrapping.
Protect Stop Date	The date and time when a managed symmetric key object is not used for applying cryptographic protection. For example, encryption or wrapping.
Deactivation Date	The date on which the object is deactivated.
Compromise Date	The date on which the object is compromised.
Revocation Reason	Indicates the reason for revoking the managed cryptographic. For example, compromised, expired, or no longer used.
Link	Identifies the target-managed cryptographic object by its unique identifier.
Name	The name to identify and locate the cryptographic object.
Object Group	A group of objects. An object might belong to more than one group of objects.
Object Type	Describes the type of the object. For example, Symmetric Key, Template, or Secret Data.
State	The state of an object as known to the key management server.
Unique Identifier	Uniquely identifies the managed object.

KMIP profiles

IBM Security Key Lifecycle Manager supports various KMIP profiles to interact with KMIP clients.

The KMIP standard contains various profiles that modify the standard for specific use cases, for example, asymmetric key storage with TLS 1.2. These profiles specify conformance to certain operations and attributes.

KMIP profiles supported by IBM Security Key Lifecycle Manager

A selected set of conformance clauses and authentication suites that when "paired together" form KMIP profiles.

Basic Discover Versions Server Profile
A profile that consists of the tuple {Discover Versions Server Conformance Clause, Basic Authentication Suite}.

Basic Baseline Server KMIP Profile
A profile that consists of the tuple {Baseline Server Conformance Clause, Basic Authentication Suite}.

Basic Secret Data Server KMIP Profile
A profile that consists of the tuple {Secret Data Server Conformance Clause, Basic Authentication Suite}.

Basic Symmetric Key Store and Server KMIP Profile
A profile that consists of the tuple {Basic Symmetric Key Store and Server Conformance Clause, Basic Authentication Suite}.

Basic Symmetric Key Foundry and Server KMIP Profile
A profile that consists of the tuple {Basic Symmetric Key Foundry and Server Conformance Clause, Basic Authentication Suite}.

Basic Asymmetric Key Store Server KMIP Profile
A profile that consists of the tuple {Basic Asymmetric Key Store Server Conformance Clause, Basic Authentication Suite}.

Basic Asymmetric Key and Certificate Store Server KMIP Profile
A profile that consists of the tuple {Basic Asymmetric Key and Certificate Store Server Conformance Clause, Basic Authentication Suite}.

Basic Asymmetric Key Foundry and Server KMIP Profile
A profile that consists of the tuple {Basic Asymmetric Key Foundry and Server Conformance Clause, Basic Authentication Suite}.

Basic Certificate Server KMIP Profile (except PEM certificate format)
A profile that consists of the tuple {Basic Certificate Server Conformance Clause, Basic Authentication Suite}.

Basic Asymmetric Key Foundry and Certificate Server KMIP Profile (except PEM certificate format)
A profile that consists of the tuple {Basic Asymmetric Key Foundry and Certificate Server Conformance Clause, Basic Authentication Suite}.

Discover Versions TLS 1.2 Authentication Server Profile
A profile that consists of the tuple {Discover Versions Server Conformance Clause, TLS 1.2 Authentication Suite}.

Baseline Server TLS 1.2 Authentication KMIP Profile
A profile that consists of the tuple {Baseline Server Conformance Clause, TLS 1.2 Authentication Suite}.

Secret Data Server TLS 1.2 Authentication KMIP Profile
A profile that consists of the tuple {Secret Data Server Conformance Clause, TLS 1.2 Authentication Suite}.

Symmetric Key Store and Server TLS 1.2 Authentication KMIP Profile
A profile that consists of the tuple {Basic Symmetric Key Store and Server Conformance Clause, TLS 1.2 Authentication Suite}.

Symmetric Key Foundry and Server TLS 1.2 Authentication KMIP Profile
A profile that consists of the tuple {Basic Symmetric Key Foundry and Server Conformance Clause, TLS 1.2 Authentication Suite}.

Asymmetric Key Store Server TLS 1.2 Authentication KMIP Profile
A profile that consists of the tuple {Asymmetric Key Store Server Conformance Clause, TLS 1.2 Authentication Suite}.

Asymmetric Key and Certificate Store Server TLS 1.2 Authentication KMIP Profile
A profile that consists of the tuple {Asymmetric Key and Certificate Store Server Conformance Clause, TLS 1.2 Authentication Suite}.

Asymmetric Key Foundry and Server TLS 1.2 Authentication KMIP Profile
A profile that consists of the tuple {Asymmetric Key Foundry and Server Conformance Clause, TLS 1.2 Authentication Suite}.

Certificate Server TLS 1.2 Authentication KMIP Profile (except PEM certificate format)
A profile that consists of the tuple {Certificate Server Conformance Clause, TLS 1.2 Authentication Suite}.

Asymmetric Key Foundry and Certificate Server TLS 1.2 Authentication KMIP Profile (except PEM certificate format)
A profile that consists of the tuple {Asymmetric Key Foundry and Certificate Store Server Conformance Clause, TLS 1.2 Authentication Suite}.

Symmetric Key Foundry Version 1.0, 1.1, and 1.2 KMIP Profile
The Symmetric Key Foundry (FIPS140) Profile is a KMIP server that responds to KMIP client requests to create symmetric keys by using FIPS 140-2 approved algorithms.

Asymmetric Key Lifecycle Version 1.0, 1.1, and V1.2 KMIP Profile
The Asymmetric Key Lifecycle Profile is a KMIP server that performs asymmetric key lifecycle operations based on requests that are received from a KMIP client.

Symmetric Key Lifecycle Version 1.0, 1.1, and 1.2 KMIP Profile
The Symmetric Key Lifecycle Profile is a KMIP server that performs symmetric key lifecycle operations based on requests that are received from a KMIP client.

Storage Array with Self-Encrypting Drives Version 1.0, 1.1, and 1.2 KMIP Profile
The Storage Array with Self-Encrypting Drives Profile is a storage array that contains self-encrypting drives operating as a KMIP client interacting with a KMIP server.

Tape Library Version 1.0, 1.1, and 1.2 KMIP Profile
The Tape Library Profile specifies the behavior of a tape library operating as a KMIP client interacting with a KMIP server.

HTTPS Message Encoding Version 1.0, 1.1, and 1.2 KMIP Profile
The Hypertext Transfer Protocol over Transport Layer Security (HTTPS) is simply the use of HTTP over TLS in the same manner that HTTP is used over TCP.

KMIP over HTTPS is simply the use of KMIP messages over HTTPS in the same manner that KMIP is used over TLS.

JSON Message Encoding Version, 1.0, 1.1, and 1.2 KMIP Profile
The JSON profile specifies the use of KMIP replacing the TTLV message encoding with a JSON message encoding.

XML Message Encoding Version 1.0, 1.1, and 1.2 KMIP Profile
The XML profile specifies the use of KMIP replacing the TTLV message encoding with an XML message encoding.

Opaque Managed Object Store Version 1.0, 1,1, and 1.2 KMIP Profile
The Opaque Managed Object Store Profile is a KMIP server that performs storage-related operations on opaque objects based on requests that are received from a KMIP client.

Suite B Version 1.0, 1.1, and 1.2 KMIP Profile
The Suite B minLOS_128 Profile describes a KMIP client interacting with a KMIP server as an information assurance product to provide a minimum level of security of 128 bits.

For more information about profiles, see the KMIP Profiles documentation.

Symmetric Key Foundry
http://docs.oasis-open.org/kmip/kmip-sym-foundry-profile/v1.0/os/kmip-sym-foundry-profile-v1.0-os.html

Asymmetric Key Lifecycle
http://docs.oasis-open.org/kmip/kmip-asym-key-profile/v1.0/os/kmip-asym-key-profile-v1.0-os.html

Symmetric Key Lifecycle
http://docs.oasis-open.org/kmip/kmip-sym-key-profile/v1.0/os/kmip-sym-key-profile-v1.0-os.html

Storage Array with Self-Encrypting Drives
http://docs.oasis-open.org/kmip/kmip-sa-sed-profile/v1.0/os/kmip-sa-sed-profile-v1.0-os.html

Tape Library
http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/os/kmip-tape-lib-profile-v1.0-os.html

Message Encoding
http://docs.oasis-open.org/kmip/kmip-addtl-msg-enc/v1.0/os/kmip-addtl-msg-enc-v1.0-os.html

Opaque Managed Object Store
http://docs.oasis-open.org/kmip/kmip-opaque-obj-profile/v1.0/os/kmip-opaque-obj-profile-v1.0-os.html

Suite B Profile
http://docs.oasis-open.org/kmip/kmip-suite-b-profile/v1.0/os/kmip-suite-b-profile-v1.0-os.html

KMIP client configuration

KMIP enables secure creation and storage of cryptographic objects on the IBM Security Key Lifecycle Manager server. You must configure client devices to connect to the server for key management operations.

KMIP objects management

The IBM Security Key Lifecycle Manager server supports Key Management Interoperability Protocol (KMIP) communication with client devices. You can create and manage cryptographic objects by using a set of operations that IBM Security Key Lifecycle Manager provides.

You can use IBM Security Key Lifecycle Manager graphical user interface for the following cryptographic object management activities:

- Registering client devices with the server
- Creating and configuring cryptographic objects for the registered client devices
- Viewing objects for the registered client devices
- Modifying client device information by adding more objects or associating a new certificate
- Deleting client devices and the associated objects from the server
- Searching for objects that are managed by the server

Registering KMIP-compliant client devices

You must register KMIP-compliant client devices with the IBM Security Key Lifecycle Manager server before the client communicates with server for key management operations.

About this task

Use **Client Dashboard** to manage client devices and its objects. You can use the dashboard to view, register, modify, and delete client devices and the associated cryptographic objects.

Associate a certificate with client device for secure communication with the server. Before you register the client, determine which of the following client device certificates to be used for communication:

- Existing client device certificate that is not in use by some other client device.
- Accepting a pending client device certificate.
- Importing a client device certificate.

You can also register client device without associating a certificate. You can later associate by selecting certificate from the pending certificate list. Click the **Pending client registration requests** link on the dashboard to select the certificate. If you accept, the certificate is imported into the database and marked as trusted. The certificate can then be used for secure communication between the client device and IBM Security Key Lifecycle Manager. You can also associate a certificate when you modify client device information.

Procedure

1. Log on to the graphical user interface by using your credentials.
2. On the Welcome page, click **Clients and Groups**.
3. On the Client Dashboard page, click **Create**.
4. In the **Client Name** field on the Register Client page, specify a name for the client.
5. Select a client certificate for secure communication with the server.

None	The client device is registered with the server without an associated client communication certificate.
Use existing client certificate not in use	Use an existing client certificate in the database, which is not in use by any other client devices. Select the certificate from the drop-down list.

Accept pending client certificate	Select a certificate from the pending certificate list that is pushed to the server from a client device and is yet to be accepted for communication with the server. Use the following steps to accept the client communication certificate and mark it as trusted when you register the client device: 1. Specify a name for the certificate in the **Certificate name** field. 2. Select a certificate from the drop-down list.
Import client certificate	Import a client device certificate into IBM Security Key Lifecycle Manager for secure communication with the client you are registering. 1. Specify a name for the certificate in the **Certificate name** field. 2. Click **Browse** to select the file and import.

6. Click **Register Client**.

What to do next

Associate cryptographic objects with the registered client. See "Creating cryptographic objects for a client device."

Creating cryptographic objects for a client device

Create and configure cryptographic objects for the KMIP-compliant client device that you registered with IBM Security Key Lifecycle Manager.

About this task

Use **Client Dashboard** to manage client devices and its objects. You can use the dashboard to view, register, modify, and delete client devices and the associated cryptographic objects.

Use the Objects with Client page to create and configure the following objects for the client device that you registered:

- Symmetric keys
- Key pairs

Procedure

1. Log on to the graphical user interface.
2. On the Welcome page, click **Clients and Groups**.
3. On the Client Dashboard page, click **Create**.
4. Create and register a client device by using the **Register Client** tab. See "Registering KMIP-compliant client devices" on page 9.
5. In the **Add Objects** tab, add and configure objects for the client device.

None	Indicates that the client device is not associated with any of the objects.

Symmetric key	Create the symmetric key object with the following configuration settings: • Number of symmetric keys for the client device. • Cryptographic algorithm that is used to create the object, such as AES or 3DES. • Bit length of the symmetric key object. • Prefix for the key. You must specify a three character value for the key by using the alphabetic characters. • Cryptographic usage mask that defines the cryptographic functions to be performed by using the object, such as Encrypt, Decrypt, Encrypt Decrypt, Sign, Sign Verify, Verify, Wrap, Unwrap, or Wrap Unwrap.
Key Pair	Create the asymmetric key pair object with the following configuration settings: • Number of key pair objects for the client device. • Cryptographic algorithm that is used to create the object, such as RSA or DSA. • Prefix for the key. You must specify a three character value for the key by using the alphabetic characters. • Cryptographic usage mask that defines the cryptographic functions to be performed by using the object, such as Encrypt, Decrypt, Encrypt Decrypt, Sign, Sign Verify, Verify, Wrap, Unwrap, or Wrap Unwrap.

6. Click **Save and Exit**. To save and add more objects, click **Save and Add More Objects**.

Modifying client device information

Use the Modify Client page to modify client device information to suit your changing needs. You can add more objects and associate a certificate with the client device that is registered with IBM Security Key Lifecycle Manager.

About this task

Use the Client Dashboard page to manage client devices and its objects. You can use the dashboard to view, register, modify, and delete client devices and the associated cryptographic objects. Your role must have a permission to the modify action.

Procedure

1. Log on to the graphical user interface.
2. On the Welcome page, click **Clients and Groups**.
3. Select a client device from the **Client Name** column.
4. Click **Modify**.
5. Alternatively, right-click a client name and then select **Modify**, or double-click a client entry.
6. On the Modify Client dialog, add more objects or associate a certificate to the client device to fit your needs.
7. Click **Save and Exit** to save the changes and exit.
8. To save and add more objects, click **Save and Add More Objects**.

Deleting client device information

You can delete client device and its objects from the IBM Security Key Lifecycle Manager database if they are no longer needed.

About this task

Use the Client Dashboard page to delete client devices and the associated cryptographic objects. Your role must have a permission to the delete action.

Before you begin, ensure that a current backup exists for the IBM Security Key Lifecycle Manager database.

Procedure

1. Log on to the graphical user interface.
2. On the Welcome page, click **Clients and Groups**.
3. Select a client device from the **Client Name** column.
4. Click **Delete**.
5. Alternatively, right-click a client name and then select **Delete**.
6. On the Confirm dialog, read the confirmation message before you delete the client device. Click **OK**. The client device and its objects are removed from the IBM Security Key Lifecycle Manager database.

Querying IBM Security Key Lifecycle Manager server for KMIP

KMIP clients can run `Query` operation with `Server Information` query function to find out whether the IBM Security Key Lifecycle Manager server is stand-alone, master, or clone.

When you run the `Query` operation, IBM Security Key Lifecycle Manager server returns `VendorInformation`, which contains `ServerType` details.

Server Type	**ServerType Information in Query Operation**
Stand-alone	No ServerType field in `VendorInformation`.
Master	ServerType=master in `VendorInformation`.
Clone	ServerType=clone in `VendorInformation`.
Multi-Master	ServerType=multi-master in `VendorInformation`.

The following sample query request and response shows how to check the type of IBM Security Key Lifecycle Manager server.

```
<RequestMessage>
    <RequestHeader>
       <ProtocolVersion>
           <ProtocolVersionMajor type="Integer" value="1"/>
           <ProtocolVersionMinor type="Integer" value="0"/>
       </ProtocolVersion>
     <MaximumResponseSize type="Integer" value="2048"/>
     <BatchCount type="Integer" value="1"/>
   </RequestHeader>
   <BatchItem>
     <Operation type="Enumeration" value="Query"/>
     <RequestPayload>
```

```
        <QueryFunction type="Enumeration" value="QueryServerInformation"/>
      </RequestPayload>
   </BatchItem>
</RequestMessage
```

```
<ResponseMessage>
 <ResponseHeader>
  <ProtocolVersion>
   <ProtocolVersionMajor type="Integer" value="1"/>
   <ProtocolVersionMinor type="Integer" value="0"/>
  </ProtocolVersion>
  <TimeStamp type="DateTime" value="2017-11-02T16:21:22+05:30"/><BatchCount type="Integer" value="
 </ResponseHeader>
 <BatchItem>
  <Operation type="Enumeration" value="Query"/>
   <ResultStatus type="Enumeration" value="Success"/>
   <ResponsePayload>
   <Operation type="Enumeration" value="Create"/>
   <Operation type="Enumeration" value="Register"/>
   <Operation type="Enumeration" value="CreateKeyPair"/>
   <Operation type="Enumeration" value="Get"/>
   <Operation type="Enumeration" value="Activate"/>
   <Operation type="Enumeration" value="AddAttribute"/>
   <Operation type="Enumeration" value="Check"/>
   <Operation type="Enumeration" value="DeleteAttribute"/>
   <Operation type="Enumeration" value="Destroy"/>
   <Operation type="Enumeration" value="GetAttributeList"/>
   <Operation type="Enumeration" value="GetAttributes"/>
   <Operation type="Enumeration" value="GetUsageAllocation"/>
   <Operation type="Enumeration" value="Locate"/>
                 <Operation type="Enumeration" value="ModifyAttribute"/>
   <Operation type="Enumeration" value="ObtainLease"/>
   <Operation type="Enumeration" value="Query"/>
   <Operation type="Enumeration" value="Revoke"/>
   <Operation type="Enumeration" value="ReKey"/>
   <Operation type="Enumeration" value="ReKeyKeyPair"/>
   <Operation type="Enumeration" value="Certify"/>
   <Operation type="Enumeration" value="ReCertify"/>
   <ObjectType type="Enumeration" value="SymmetricKey"/>
   <ObjectType type="Enumeration" value="Template"/>
   <ObjectType type="Enumeration" value="SecretData"/>
   <ObjectType type="Enumeration" value="PrivateKey"/>
   <ObjectType type="Enumeration" value="PublicKey"/>
   <ObjectType type="Enumeration" value="Certificate"/>
   <VendorIdentification type="TextString" value="SKLM 3.0.0.0 KMIP 1.3 BUILD 201711071556
                   KMIP_SSL_TIMEOUT 5 SERVER_TYPE=Multi-Master
                   CLUSTER_DETAILS=WIN-764BULETAOD:5696:0,master2:5696:0,WIN-RJF3F58VAJ6:5696:0
                   HADR_STATUS= master2:CONNECTED WIN-RJF3F58VAJ6:CONNECTED,WIN-764BULETAOD:CONNE
   <ServerInformation/>
   </ResponsePayload>
 </BatchItem>
</ResponseMessage>
```

CLUSTER_DETAILS

Indicates host names of the masters in the cluster, for example, WIN-764BULETAOD, master2, WIN-RJF3F58VAJ6.

Indicates KMIP port number of IBM Security Key Lifecycle Manager master servers, for example 5696.

Indicates Non-HADR status of IBM Security Key Lifecycle Manager master servers.

HADR_STATUS

Indicates HADR status of IBM Security Key Lifecycle Manager master servers. Possible values are as follows.

CONNECTED
DISCONNECTED
NOT_IN_CLUSTER

HADR_STATUS_CODE
Indicates HADR status of IBM Security Key Lifecycle Manager Multi-Master cluster. Possible values are as follows.

`0` All instances in the cluster are connected.
`-1` Few instances in the cluster are connected.
`-2` None of the instances in the cluster are connected.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law :

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability
These terms and conditions are in addition to any terms of use for the IBM website.

Personal use
You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use
You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at http://www.ibm.com/legal/copytrade.shtml.

Adobe, Acrobat, PostScript and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Index