Example of using endpoints to manage the Key Management Service integration

Version 64.2 authored by superadmin on 2025/02/25 07:28:00

Table of Contents

Use the Hold Your Own Key (HYOK) REST APIs to securely manage encryption keys and integrate with the HYOK providers (AWS and OCI). These APIs support key operations, such as creating and updating KMS configurations and performing DEK rotation for enhanced security.

Related topics

Learning-about-the-REST-API

Endpoints-in-AR-REST-API

Enhancing-data-security-by-managing-your-own-encryption-keys

Endpoint to check the status of the existing Data Encryption Key (DEK)

Use the `check` endpoint to check if HYOK is enabled or not.

URL qualifier	/api/arsys/v1.0/dek/check
Method	GET
Header	

Header	Value	Type
X-Requested-By	XMLHttpRequest	Default

Request body	This API does not require a request body.
Parameters	This API does not accept any query or path parameters.
Returns	The response provides details about HYOK.

Success

Returns the DEK status details.

Example Response:

```
{
    "hyokenabled": true,
    "hyokprovider": "aws",
    "isHYOKEnabled": true,
    "HYOKProvider": "aws"
}
```

Failure

Returns an error message with the reason for failure.

Example Response:

```
{
    "hyokprovider": "No-Provider-Configured",
    "hyokenabled": false,
    "isHYOKEnabled": false,
    "HYOKProvider": "No-Provider-Configured"
}
```

Example

The API details for checking whether HYOK is enabled or not is shown in the following example:

Request URL

```
https://example.com/api/arsys/v1.0/dek/check
```

Request header

```
X-Requested-By: XMLHttpRequest
Cookie: AR-JWT=eyJhbGeyJhbGciOiJIUzI1NiJ9.abc123ExampleSignature; _cacheId=CacheId; route=RouteId \
```

Request Body
(No data is required for this API)

Response

```
{
    "hyokenabled": true,
    "hyokprovider": "aws",
    "isHYOKEnabled": true,
    "HYOKProvider": "aws"
}
```

Endpoint to create the OCI provider connection details

Use the `kmsconfig_create` endpoint to create the Key Management Service (KMS) for OCI. The endpoint requires the necessary details, such as access credentials, master key ID, and region, to establish communication with OCI KMS for encryption and key management operations.

URL qualifier	/api/arsys/v1.0/kmsconfig/create
Method	POST

Header

Header	Value	Type
X-Requested-By	XMLHttpRequest	Default
Content-type	multipart/form-data	text

Parameters

Name	Value	Type	Description
provider	oci	Text	Specifies the HYOK provider.
tenantId	tenantid	Text	Specifies the OCI tenant ID.
userId	userid	Text	Specifies the OCI user ID.
fingerprint	fingerprint	Text	Fingerprint of the OCI user's private key.
cryptoEndpoint	cryptoendpoint	Text	Endpoint for cryptographic operations.
mgmtEndpoint	mgmtendpoint	Text	Endpoint for management operations.
masterKeyId	master key id	Text	Specifies the master key ID.
privateKey		File	File input for uploading the private key associated with the API key.

For more information, see Endpoints-in-AR-REST-API.

Returns

Success

Returns the Success Response (HTTP 200) in JSON format. Example:

```
{
"status": "success",
"message": "KMS Configuration created Successfully"
}
```

Failure

Returns an error message with the reason for failure.

```
{
    "messageType": "ERROR",
    "messageText": "Cannot create KMS configuration",
    "messageAppendedText": "HYOK is already enabled and provider is configured",
    "messageNumber": 9035
}
```

Example

The API details for creating a new KMS configuration by using the POST method is shown in the following example:

Request URL

```
http://localhost:8008/api/arsys/v1.0/kmsconfig/create
```

Request header

```
X-Requested-By: XMLHttpRequest
Content-Type: multipart/form-data
Cookie: AR-JWT=eyJhbGciOiJIUzI1NiJ9.exampleTokenValue123456
```

Request body (Form data)

```
{
  "provider": "oci",
  "tenantId": "ExampleTenantId123456789",
  "userId": "ExampleUserId123456789",
  "fingerprint": "aa:bb:cc:dd:ee:ff:11:22:33:44:55:66:77:88:99:00",
  "cryptoEndpoint": "https://example-crypto.kms.us-ashburn-1.oraclecloud.com",
  "mgmtEndpoint": "https://example-management.kms.us-ashburn-1.oraclecloud.com",
  "masterKeyId": ocid1.key.oc1.iad.exampleKeyId123456",
  "privateKey": "@/C:/path/to/example-private-key.pem"
}
```

Response

```
{
"status": "success",
"message": "KMS Configuration created Successfully"
}
```

Endpoint to create the AWS provider connection details

Use the `kmsconfig_create` endpoint to retrieve the Key Management Service (KMS) for AWS. It requires the necessary details, such as access credentials, master key ID, and region, to establish communication with AWS KMS for encryption and key management operations.

URL qualifier	/api/arsys/v1.0/kmsconfig/create
Method	POST

Header

Header	Value	Type
X-Requested-By	XMLHttpRequest	Default
Content-type	multipart/form-data	text

Parameters

Name	Value	Type	Description
provider	aws	Text	Specifies the HYOK provider.
accessKey	accessKey	Text	Specifies the AWS access key ID for authentication

secretKey	secretKey	Text	Specifies the AWS secret access key for authentication.
masterKeyId	master key id	Text	Specifies the identifier for the master key in AWS KMS.

For more information, see Endpoints-in-AR-REST-API.

Returns

Success

Returns the Success Response (HTTP 200) in JSON format. Example:

```
{
"status": "success",
"message": "KMS Configuration created Successfully"
}
```

Failure

Returns an error message with the reason for failure.

```
{
    "messageType": "ERROR",
    "messageText": "Cannot create KMS configuration",
    "messageAppendedText": "HYOK is already enabled and provider is configured",
    "messageNumber": 9035
}
```

Example

The API details for creating a new KMS configuration by using the POST method is shown in the following example:

Request URL

```
http://localhost:8008/api/arsys/v1.0/kmsconfig/create
```

Request header

```
X-Requested-By: XMLHttpRequest
Content-Type: multipart/form-data
Cookie: AR-JWT=eyJhbGciOiJIUzI1NiJ9.abc123TokenValue; _cacheId=CacheID; onbmc_pool=PoolID; route=RouteID' \
```

Request body (Form data)

```
{
provider="aws"
accessKey="AKIAEXAMPLEKEY12345"
secretKey="EXAMPLESECRETKEY12345"
masterKeyId="mrk-Examplekeyid123456789"
region="ap-south-1"
}
```

Response

```
{
"status": "success",
"message": "KMS Configuration created Successfully"
}
```

Endpoint to update the OCI provider connection details

Use the `kmsconfig_update` endpoint to update the configuration details for connecting to OCI Key Management Service (KMS). It allows users to provide specific connection parameters, such as tenant ID, user ID, endpoints, and keys, required to establish secure communication with the OCI KMS.

URL qualifier	/api/arsys/v1.0/kmsconfig/update
Method	POST

Header

Header	Value	Type
X-Requested-By	XMLHttpRequest	Default
Content-type	multipart/form-data	text

Parameters

Name	Value	Type	Description

provider	oci	Text	Specifies the HYOK provider.
tenantId	tenantid	Text	Specifies the OCI tenant ID.
userId	userid	Text	Specifies the OCI user ID.
fingerprint	fingerprint	Text	Fingerprint of the OCI user's private key.
cryptoEndpoint	cryptoendpoint	Text	Endpoint for cryptographic operations.
mgmtEndpoint	mgmtendpoint	Text	Endpoint for management operations.
masterKeyId	master key id	Text	Specifies the master key ID.
privateKey		File	File input for uploading the private key associated with the API key.

For more information, see Endpoints-in-AR-REST-API.

Returns

Success

Returns the Success Response (HTTP 200) in JSON format. Example:

```
{
"status": "success",
  "message": "KMS Configuration updated Successfully"
}
```

Failure

Returns an error message with the reason for failure.

```
{
    "messageType": "ERROR",
    "messageText": "Cannot update KMS configuration",
    "messageAppendedText": "HYOK is not configured, create KMS configuration first",
    "messageNumber": 9036
}
```

Example

The API details for updating the configuration details for connecting to an OCI KMS by using the POST method is shown in the following example:

Request URL

```
http://localhost:8008/api/arsys/v1.0/kmsconfig/update
```

Request header

```
X-Requested-By: XMLHttpRequest
Content-Type: multipart/form-data
Cookie: AR-JWT=eyJhbGciOiJIUzI1NiJ9.abc123TokenValue; _cacheId=CacheId; onbmc_pool=PoolId; route=RouteId' \
```

Request body (Form data)

```
{
  "provider": "oci",
  "tenantId": "ExampleTenantId123456789",
  "userId": "ExampleUserId123456789",
  "fingerprint": "aa:bb:cc:dd:ee:ff:11:22:33:44:55:66:77:88:99:00",
  "cryptoEndpoint": "https://example-crypto.kms.us-ashburn-1.oraclecloud.com",
  "mgmtEndpoint": "https://example-management.kms.us-ashburn-1.oraclecloud.com",
  "masterKeyId": ocid1.key.oc1.iad.exampleKeyId123456",
  "privateKey": "@/C:/path/to/example-private-key.pem"
}
```

Response

```
{
"status": "success",
  "message": "KMS Configuration updated Successfully"
}
```

Endpoint to update the AWS provider connection details

Use the `kmsconfig_update` endpoint to update the configuration details for connecting to Amazon Web Services (AWS) Key Management Service (KMS). It requires the necessary details, such as access credentials, master key ID, and region, to establish communication with AWS KMS for encryption and key

management operations.

URL qualifier	/api/arsys/v1.0/kmsconfig/update
Method	POST

Header

Header	Value	Type
X-Requested-By	XMLHttpRequest	Default
Content-type	multipart/form-data	text

Parameters

Name	Value	Type	Description
provider	aws	Text	Specifies the HYOK provider.
accessKey	accessKey	Text	Specifies the AWS access key ID for authentication.
secretKey	secretKey	Text	Specifies the AWS secret access key for authentication.
masterKeyId	master key id	Text	Specifies the identifier for the master key in AWS KMS.
region	region	File	File input for uploading the private key associated with the API key.

For more information, see Endpoints-in-AR-REST-API.

Returns

Success

Returns the Success Response (HTTP 200) in JSON format. Example:

```
{
"status": "success",
  "message": "KMS Configuration updated Successfully"
}
```

Failure

Returns an error message with the reason for failure.

```
{
      "messageType": "ERROR",
      "messageText": "Cannot update KMS configuration",
      "messageAppendedText": "HYOK is not configured, create KMS configuration first",
      "messageNumber": 9036
 }
```

Example

The API details for updating the configuration details for connecting to an AWS KMS by using the POST method is shown in the following example:

Request URL

```
http://localhost:8008/api/arsys/v1.0/kmsconfig/update
```

Request header

```
X-Requested-By: XMLHttpRequest
Content-Type: multipart/form-data
Cookie: AR-JWT=eyJhbGciOiJIUzI1NiJ9.abc123TokenValue; _cacheId=CacheId; onbmc_pool=PoolId; route=RouteId' \
```

Request body (Form Data)

```
provider="aws"
accessKey="AKIAEXAMPLEKEY12345"
secretKey="EXAMPLESECRETKEY12345"
masterKeyId="mrk-Examplekeyid123456789"
region="ap-south-1"
```

Response

```
{
"status": "success",
  "message": "KMS Configuration updated Successfully"
}
```

Endpoint to rotate the DEK

Use the `rotate` API to rotate the Data Encryption Key (DEK) for enhanced security. When invoked, the API generates a new DEK through the configured HYOK provider and starts using it immediately for encryption operations. This method ensures that data is encrypted with the latest key, mitigating potential risks from key exposure.

URL qualifier	/api/arsys/v1.0/dek/rotate
Method	POST
Header	

Header	Value	Type
X-Requested-By	XMLHttpRequest	Default

Parameters	The rotate API does not require any input in the body. The body is empty.
Returns	

Success

If the DEK rotation is successful, the API responds with a success message.

```
{
    "message": "Kek rotated successfully"
}
```

Failure

If the rotation fails due to configuration issues, connectivity errors, or other reasons, an error response is returned.

```
{
    "messageType": "ERROR",
    "messageText": "HYOK is not enabled",
    "messageAppendedText": "HYOK is not Enabled",
    "messageNumber": 9039
}
```

Example

The API details for rotating the Data Encryption Key (DEK) using the POST method is shown in the following example:

Request URL

```
http://localhost:8008/api/arsys/v1.0/dek/rotate
```

Request header

```
X-Requested-By: XMLHttpRequest
Cookie: AR-JWT=eyJhbGciOiJIUzI1NiJ9.abc123Signature; _cacheId=CacheId; route=RouteId \
```

Request body
(No data is required for this API)

Response

```
{
    "message": "Kek rotated successfully"
}
```

Troubleshooting the KMS APIs

Here are some common issues you might encounter when using the HYOK REST APIs and the steps to resolve them:

Unable to create the KMS configuration

Issue

If you receive the following error messages, it means that the HYOK provider has already been set up:

- `Cannot create KMS configuration`
- `HYOK is already enabled and provider is configured`

Resolution

- Ensure that you are using the correct provider (AWS or OCI).
- Verify that the provider configuration exists by using the `kmsconfig_get` endpoint API.
- This error occurs if `arkmsconfig` exists in the Kubernetes secret, but the KMS settings are missing. To resolve the issue, delete the `arkmsconfig` entry from the Kubernetes secret.

Invalid master key ID

Issue

If the master key ID provided is incorrect, the API returns an error indicating that the key ID is not valid.

Resolution

Double-check the key ID against your KMS console and ensure it is correctly configured.