Encryption Service

Introduction

Encryption Service is used to secure the data.

Features

Encryption Service offers following features:

1. Encrypt - The service will encrypt the data based on given input parameters and data to be encrypted. The encrypted data will be mandatorily of type string.
2. Decrypt - The decryption will happen solely based on the input data (any extra parameters are not required). The encrypted data will have identity of the key used at the time of encryption; the same key will be used for decryption.
3. Sign - Encryption Service can hash and sign the data which can be used as unique identifier of the data. This can also be used for searching given value from a data store.
4. Verify - Based on the input sign and the claim, it can verify if the given sign is correct for the provided claim.
5. Rotate Key - Encryption Service supports changing the key used for encryption. The old key will still remain with the service which will be used to decrypt old data. All the new data will be encrypted by the new key.

Supplementary Java Library

The Java library is made to ease encrypting and decrypting JSON objects. At the time of encryption / decryption it will process only those attributes which are configured to be encrypted / decrypted. At the time of decryption, based on the roles of the logged in user it will decrypt / mask the attributes which are granted to that role.

Features of the library

1. Encrypt JSON Object - The library will have the configuration about the list of attributes to be encrypted for the given input object. It will filter the JSON object by these attributes. The new created JSON object which has only the values to be encrypted will be forwarded to the

encryption service by RestAPI call. The returned object of this call will have all the encrypted values. The attributes that were previously filter out are added to this encrypted object.

2. Decrypt JSON Object - The library will perform filtering and recreating the JSON object as described above for encryption. At the time of decryption, it will also check the roles of the logged in user and accordingly get a list of attributes to be decrypted. These attributes may have different access types like plain, mask or none. In case of plain access type the data will be visible to the user in clear. For mask access type, the data will be masked according to a predefined masking technique. In case of none access type; the data will be replaced with a meaningful message like "Confidential Information".
3. Audit - The library provides a function which will be used to audit any decryption request. It pushes the given audit data onto a kafka topic.

Introduction

Encryption Service is used to secure the data.

Features

Encryption Service offers following features:

1. Encrypt - The service will encrypt the data based on given input parameters and data to be encrypted. The encrypted data will be mandatorily of type string.
2. Decrypt - The decryption will happen solely based on the input data (any extra parameters are not required). The encrypted data will have identity of the key used at the time of encryption; the same key will be used for decryption.
3. Sign - Encryption Service can hash and sign the data which can be used as unique identifier of the data. This can also be used for searching given value from a data store.
4. Verify - Based on the input sign and the claim, it can verify if the given sign is correct for the provided claim.
5. Rotate Key - Encryption Service supports changing the key used for encryption. The old key will still remain with the service which will be used to decrypt old data. All the new data will be encrypted by the new key.

Supplementary Java Library

The Java library is made to ease encrypting and decrypting JSON objects. At the time of encryption / decryption it will process only those attributes which are configured to be encrypted / decrypted. At the time of decryption, based on the roles of the logged in user it will decrypt / mask the attributes which are granted to that role.

Features of the library

1. Encrypt JSON Object - The library will have the configuration about the list of attributes to be encrypted for the given input object. It will filter the JSON object by these attributes. The new created JSON object which has only the values to be encrypted will be forwarded to the encryption service by RestAPI call. The returned object of this call will have all the encrypted values. The attributes that were previously filter out are added to this encrypted object.
2. Decrypt JSON Object - The library will perform filtering and recreating the JSON object as described above for encryption. At the time of decryption, it will also check the roles of the logged in user and accordingly get a list of attributes to be decrypted. These attributes may have different access types like plain, mask or none. In case of plain access type the data will be visible to the user in clear. For mask access type, the data will be masked according to a predefined masking technique. In case of none access type; the data will be replaced with a meaningful message like "Confidential Information".
3. Audit - The library provides a function which will be used to audit any decryption request. It pushes the given audit data onto a kafka topic.