**S4 Client**

This application is a sample application that consumes S4 Security Service.

In this sample application, you can try most of the functionalities that are offered by S4.

You will not be able to use S4 service directly because this is not one for team members to use directly, it is a service to be consumed by other applications.

Applications access is managed by the organization using Application Roles. And roles are mapped to different encryption keys that are rotatable.

For more information you can check out the documentation.

As any other client application, it should meet these requirements:

1. It should provide the token and the right scope to consume the API.
2. The token should include the roles of the API.
3. S4 admin should assign these roles to the right keys to be used.

## What is S4:

S4 stands for Secure Shared Storage Service, which is a restful API that provides a mechanism to share files between team (not individuals) in a secure way.

It can also be used as an encryption service.

<https://function-s4-prod-eastus-001.azurewebsites.net/api/s4?blobName=&keyName=Key2&keyVersion=&strategy=EncryptOnly>

# S4 Service Documentation

Basically, there are 5 main functionalities of S4 that you can use this client for, they are detailed below:

You can, however, create your own client application but this application needs to have the right Roles in order to be able to access the

## Encrypt Only Scenario

Use this scenario if you don’t want to store information, you just want it encrypted in a managed way.

This is an http POST.

Your request parameters look like so:

|  |  |  |
| --- | --- | --- |
| Parameter | Condition | Description |
| BlobName | NA |  |
| KeyName | Recommended | The name of the encryption key, you can select, each authorized application (client) will be authorized to use one or more keys by the system admin.  If not provided, the system will pick one for you but if you do not have access to it , the operation will fail. |
| KeyVersion | Optional | If not supplied, the system will use the latest version of the key selected. |
| Strategy | Required | This should be “EncryptOnly” |
| Content | Required in the body | The content you want to secure and encrypt. |

You will receive a response with multiple fields, as so:

|  |  |
| --- | --- |
| Field | Description |
| Content | This will be the encrypted text of the content provided in the request |
| KeyName | This is the name of the key that was provided |
| KeyVersion | The version used for this encryption, you need to keep this value for your decryption as it is used for extra validation. |

## Decrypt Only Scenario

You use this when you have encrypted data from the previous call, and you need to make sure your client has access to the encryption key used. If not, you need to contact S4 administrator.

This is also a POST to the S4 service with these parameters.

|  |  |  |
| --- | --- | --- |
| Parameter | Condition | Description |
| BlobName | NA |  |
| KeyName | Required | The name of the encryption key, you can select, each authorized application (client) will be authorized to use one or more keys by the system admin. |
| KeyVersion | Required | If not supplied, the system will use the latest version of the key selected. |
| Strategy | Required | This should be “DecryptOnly” |
| Content | Required in the body | The content you want to decrypt. It should be the same content that you received upon encryption |

You will receive the content in simple text in the http response.

|  |  |
| --- | --- |
| Field | Description |
| Content | This will be the encrypted text of the content provided in the request |

## Store data

This operation is also a POST.

<https://function-s4-prod-eastus-001.azurewebsites.net/api/s4?blobName=A&keyName=Key1&keyVersion=3808151669ed4e1fa3bc4b44617f4e27&strategy=None>

|  |  |  |
| --- | --- | --- |
| Parameter | Condition | Description |
| BlobName | Recommended | If not supplied, the system will generate one for you, currently the service does not support override of the blobName, make sure this does not exist or delete it before you use it again. |
| KeyName | Recommended | The name of the encryption key, you can select, each authorized application (client) will be authorized to use one or more keys by the system admin.  If not provided, the system will pick one for you but if you do not have access to it , the operation will fail. |
| KeyVersion | Optional | If not supplied, the system will use the latest version of the key selected. |
| Strategy | NA |  |
| Content | Required in the body | The content you want to secure and encrypt. |

|  |  |
| --- | --- |
| Field | Description |
| BlobName | It will be the same one that you provided, or the system generated name that you will need to provide when “GetData” |
| KeyName | This is the name of the key that was provided |
| KeyVersion | The version used for this encryption; you need to keep this value for your decryption as it is used for extra validation. |

## GetData

This is a simple http GET, where you provide the following parameters in the request:

|  |  |  |
| --- | --- | --- |
| Parameter | Condition | Description |
| BlobName | Required | The name of the blob, must match the name you received fron the response of the StoreData operation. |
| KeyName | Recommended | The name of the encryption key, you can select, each authorized application (client) will be authorized to use one or more keys by the system admin.  If not provided, the system will pick one for you but if you do not have access to it , the operation will fail. |
| KeyVersion | Required | If not supplied, the system will use the latest version of the key selected. |
| Strategy | NA |  |
| NA |  |  |

And In response you will receive only one field. or an error message indicating something went wrong, like Unauthorized. Or NotFound etc.

|  |  |
| --- | --- |
| Field | Description |
| Content | This will be the encrypted text of the content provided in the request |

## Delete Data

Use this request when you want to delete the stored document and totally remove it from the secure storage.

This is an http DELETE and it does require the blob name

|  |  |  |
| --- | --- | --- |
| Parameter | Condition | Description |
| BlobName | Required | This is the file you need to remove from the storage |
| KeyName | Required | The name of the encryption key, you can select, each authorized application (client) will be authorized to use one or more keys by the system admin.  If not provided, the system will pick one for you but if you do not have access to it , the operation will fail. |
| KeyVersion | Required | If not supplied, the system will use the latest version of the key selected. |
| Strategy | NA | This should be “EncryptOnly” |
| Content | NA | The content you want to secure and encrypt. |

On a successful operation you will receive either “**Deleted Successfully**” or “**Moved Permanently**”.