

# Symmetric Searchable Encryption component

API specification version 0.1.0

Component name: SSE – Symmetric Searchable Encryption

Component deployment name: sse

## Changelog

### 2020-02-26: Version 0.1.0

Initial release with Javascript APIs for upload and search data

## Table of Contents

Changelog.....	1
2020-02-26: Version 0.1.0 .....	1
Terminology .....	1
Introduction .....	1
API description.....	2
Upload data: function <code>uploadData(data,file_id,K1,K2)</code> .....	2
Search data: function <code>search(data,K1,K2)</code> .....	2

## Terminology

Terminology/ Abbreviation	Explanation
End-user	User who uploads/ searches data
SSE server	Server which stores encrypted data
Trusted Authority	Server which stores metadata necessary for upload/ search encrypted data

## Introduction

This API specification covers APIs relevant to uploading and search data, which are implemented in Javascript. Data upload is the process that a user chooses to send data, i.e. Json object, to SSE server in cloud. Data search is the process when a user wishes to search for the stored encrypted data in SSE server by providing a Json object by template. The following section describes specification of the Javascript library functionalities which supports the above processes.

## API description

### Upload data: **function** uploadData(data,file\_id,K1,K2)

This API allows a user to encrypt a Json object, then send its ciphertext to SSE server. It currently supports the following Json object format:

- JSON objects are surrounded by curly braces {}.
- JSON objects are written in key/value pairs.
- Keys must be strings, and values can be string, number. It does not support array and object type for values.
- Keys and values do not contain the vertical slash symbol, i.e. "|" (Because the vertical slash is used as string denominator in the implementation).
- Keys and values are case-sensitive
- Keys and values are separated by a colon.
- Each key/value pair is separated by a comma.

#### Parameters:

Name	Type	Description
data	Json object	Data to be uploaded, which is a JSON object
file_id	String	File identifier, which must be unique string
K1	String	Symmetric key, which is shared between users and TA
K2	string	Symmetric key, which is shared among users, who are allowed to upload/ search data

#### Example:

```
uploadData({firstname: "David", lastname: "White", age: 25},"id1","key1","key2")
```

#### Response

### Search data: **function** search(data,K1,K2)

Search for encrypted data in SSE server by providing a search content. SSE server will return encrypted files which contain the searched keyword.

The search content is a Json object, which follows the following format:

- JSON objects are surrounded by curly braces {}.
- JSON objects are written in one key/value pair.

- Key is "keyword", and value is a combined string of an attribute and its value separated by the vertical slash symbol, i.e. "|". For instance, if a user wishes to search for "firstname=David", the value will be "firstname|David".
- Key and value are case-sensitive

Example:

```
{
  "keyword": "firstname|David"
}
```

#### Parameters:

Name	Type	Description
data	Json	Searched data, which is a Json object
K1	String	Symmetric key, which is shared between users and TA
K2	String	Symmetric key, which is shared among users, who are allowed to upload/ search data

#### Example:

```
search({"keyword": "firstname|David"}, "key1", "key2")
```

#### Response

Returned type	Description
Json object	Json object contains the number of found objects, and their content. {count: <number of found objects>, objects: <Array of Json objects, which contain decrypted data>}