MyData Service Linking Specification

Notice

MyData Architecture defines the operations and APIs between the Operational Roles (Operator, Source, Sink etc.). Any descriptions or figures of the role's internal structure or operations are for illustrative purposes only.

1. Introduction

This document specifies MyData Service Linking.

This document is part of the MyData architecture release 1.2.1. The reader is assumed to be familiar with the 'MyData Architecture - Consent Based Approach for Personal Data Management' document and with the parallel technical specification documents available at https://hiit.github.io/mydata-stack/.

Known deficiencies in this release: limited error handling and HTTP error messages. These will be part of the next release of this document.

1.1 Definitions

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

1.2 Diff/revision history

In release 1.2.1:

- timestamp format corrected
- MyData Service Link Record details corrected
- MyData Service Link Status Record details corrected

1.3 Terminology

Key terminology used in this specification is defined in the Glossary of 'MyData Architecture - Consent Based Approach for Personal Data Management' release 1.2.1 available at https://hiit.github.io/mydata-stack/.

Service Linking [interaction] Account Owner's act of linking a service (Source or Sink) to their MyData Account. As the result the Service Linking status and parameters are documented within a digital machine-readable record, called a Service Link Record.

Service Link Record (SLR) is the outcome of a successful Service Linking. It documents in machine readable form the terms and scope of the agreement between the Account Owner and a single Source or Sink. Service Link Records are stored in the MyData Account.

Service Link Status Record (SSR) is a record MyData Operator sends to a service when status of a Service Link changes. Service MUST store these records for future use.

Surrogate ID is a pseudonym that associates Account Owner's MyData Account to his / her account at the service being linked. This ID is meaningful only to Operator and to the service that generated it. It is used in communication between these two parties whenever they need to unambiguously refer to

a specific Account Owner's MyData Account (messages from service to Operator), or to a specific user account at the service (messages from Operator to service).

1.4 Formats

In MyData Architecture, all data records and their respective digital signatures exchanged between actors are expressed using Javascript Object Notation (JSON). Digital signatures are expressed as JSON Web Signature (JWS)-structures and cryptographic keys as JSON Web Key (JWK)-structures.

In this document, JSON definitions of the data records are presented without JWS structures.
All Timestamps are in UTC in the NumericDate format as defined in [RFC7519].

2 MyData Service Link Model

To be able to manage access to their data, the Account Owner first has to attach the related service to the MyData Account. MyData ***Service Linking*** means the act of *adding a service* (a Source or a Sink) to specific Account Owner's MyData Account.

2.1 Service Link Record

A successful Service Linking results in a ***Service Link Record (SLR)*** both stored in the MyData Account and sent to the related service.

SLR also defines the keys used to sign the individual's MyData Consent Records. A valid SLR is required before any consents can be issued or used within the MyData ecosystem.

2.2 Service Link Record Lifecycle

As the Service Link Record is a signed, immutable object, the status of the link is indicated by a separate ***Service Link Status Record (SSR)*** both stored in the MyData Account and sent to the related service.

There are only two available statuses: *Active* and *Removed*. A newly created Service Link is always *Active*. If, for some reason, a link is revoked, the status is changed to *Removed*, after which it no longer can be changed to *Active*.

A Service Link is only valid with an Active Status Record.

2.3 Binding Between Service and Account Owner

For Account Owner to link and use a service, they have to have an existing account with the service before completing the Service Linking process. At the latest, they have to complete the sign-up process during the Service Linking.

In the current version of this specification, we assume that Account Owner already has an existing account at the service being linked.

During the linking process, the service creates a **Surrogate ID** that is a pseudonym that associates Account Owner's MyData Account to his / her account at the service being linked. This ID is meaningful only to Operator and to the service that generated it. It is used in communication between these two parties whenever they need to unambiguously refer to a specific Account Owner's MyData Account (messages from service to Operator), or to a specific user account at the service (messages from Operator to service).

3 Service Linking Transactions

Service Linking has four transactions:

- creating a link, removing a link
- requesting a copy of an existing Service Link Record from the Operator
- requesting a copy of the Service Link Status Record from the Operator

3.1 Creating a Service Link

Motivation

Account Owner wants to manage access to data at a service through the Operator. First action required is attaching the related service to the MyData Account.

Prerequisites:

- Service that is to be linked is registered into Service Registry.
- Account Owner starts linking process at the Operator. Account Owner MAY have been redirected to Operator from the service to be linked. Some services, e.g. governmental services, could in the future be linked automatically to an Account, but normally Account Owner initiates the linking.

Process: (steps refer to figure 3.2)

- *Step 1:* Operator fetches information needed for linking from Service Registry.
- *Step 2:* Operator requests Surrogate ID and in case of a Sink service being linked, the public part of Proof-of-Possession Key[1] from service. Service returns the requested items to Operator after it has authenticated Account Owner and Account Owner has confirmed Service Linking.
- *Step 3:* Operator constructs a Service Link Record (SLR), Account Owner signs the SLR, Operator sends a copy of SLR to the Service. Service signs SLR and sends it back to Operator.
- *Step 4:* Operator verifies Service's signature, creates initial Service Link Status Record (SSR), stores SLR and SSR into Account, and sends SLR and SSR to the Service.

Outcome:

- Service is linked to Account Owner's MyData Account.
- SLR and SSR are created

Additional info:

Operator MUST deliver copy of Service Link Record and Service Link Status Record to the service after it has been created.

A simplified flow is shown in Figure 3.1 and a more detailed flow is shown in Figure 3.2.

[1]Sink's Proof-of-Possession Key is later used for signing Data requests. Operator needs to store the returned public key (or keys, if a Sink uses several) for further use.

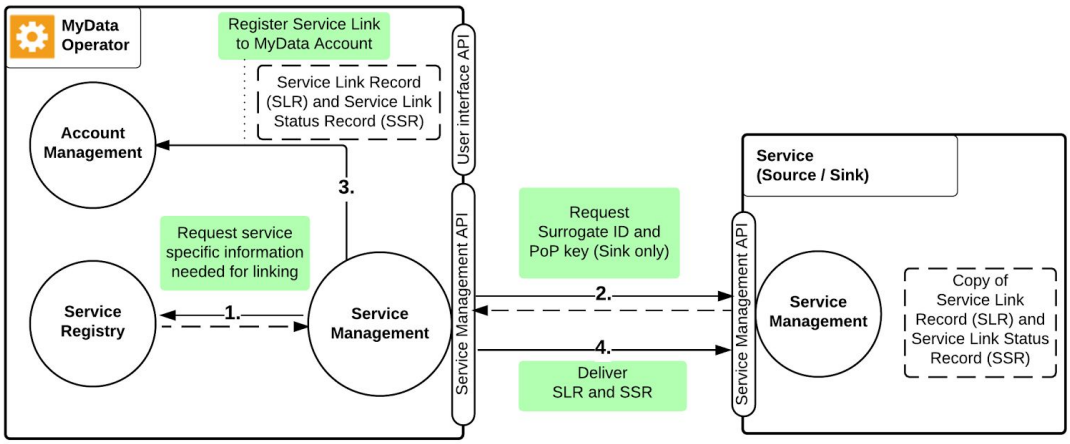

Figure 3.1: Service Linking flow

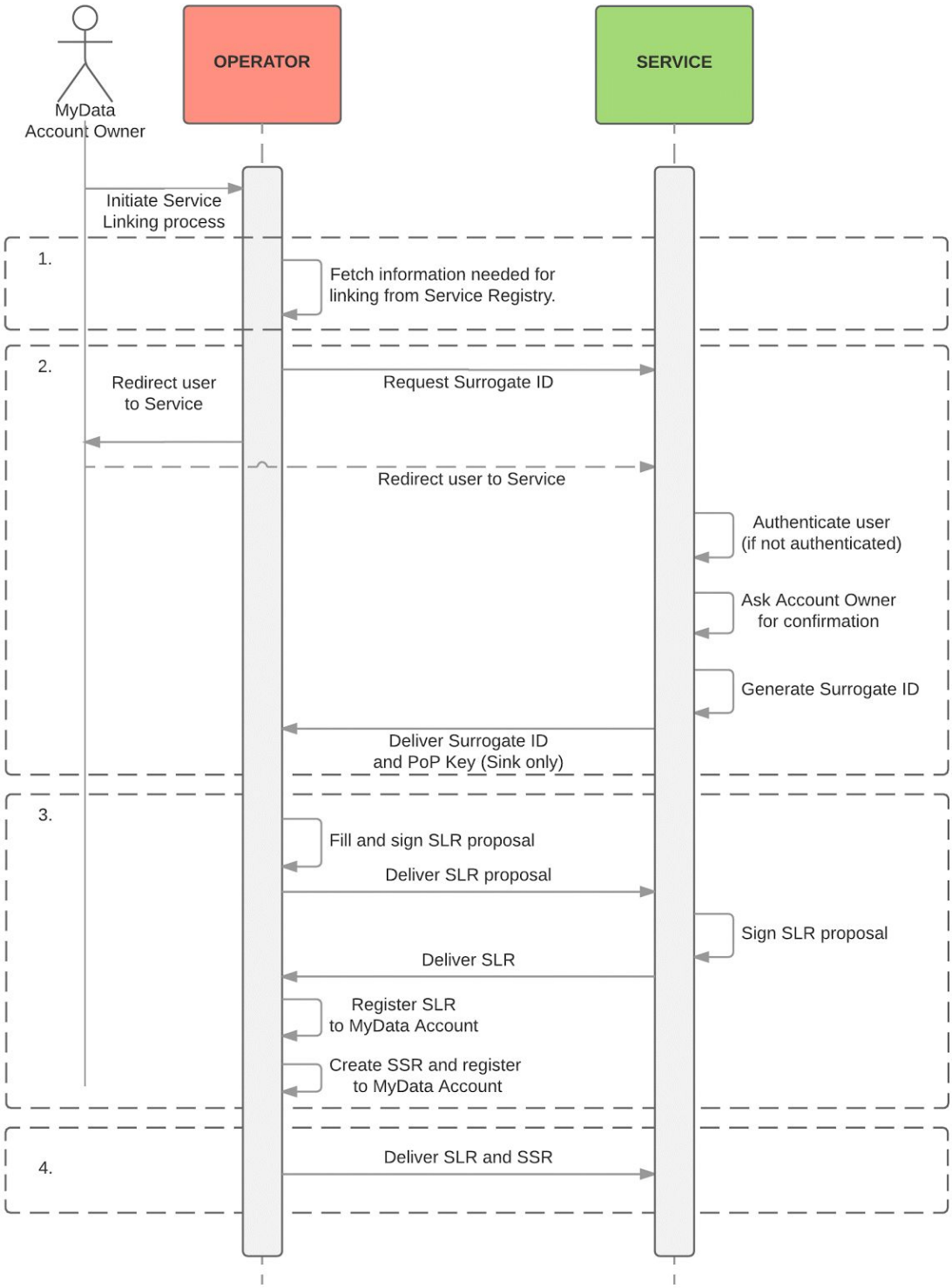

Figure 3.2: Flow overview of Service Linking

3.2 Removing a Service Link

Motivation
Service Link removal process is initiated when either a) Account Owner wants to remove a Service from MyData Account, b) Service deregisters from the Operator, or c) Account Owner removes an account at the Service and there is no need to keep the Service Link, in which case it's Service's duty to remove the unnecessary Service Link.

Prerequisites: Service Link exists

Process:

- Operator is notified that Service Link is to be removed.
- Operator creates new SSR with *Removed* state.
- Operator sends SSR to Service.
- Operator sets all related Consent Records to Disabled state.

Outcome: Service Link is set to Removed state. All Consent Records made under this Service Link MUST be set to "Disabled" state.

Additional info:
Either party - Account Owner or Service that has been linked - MAY initiate the Service Link removal. Operator MUST notify the impacted services about the removal.
Operator MAY wait for confirmation from the Service.

If removal is initiated by the Service, it MUST NOT remove the Service Link without notifying the Operator. Service SHOULD wait for confirmation from Operator before removing the Service Link. If Service doesn't receive confirmation from the Operator it MAY initiate Service Linking removal again.

A simplified flow is shown in Figure 3.3.

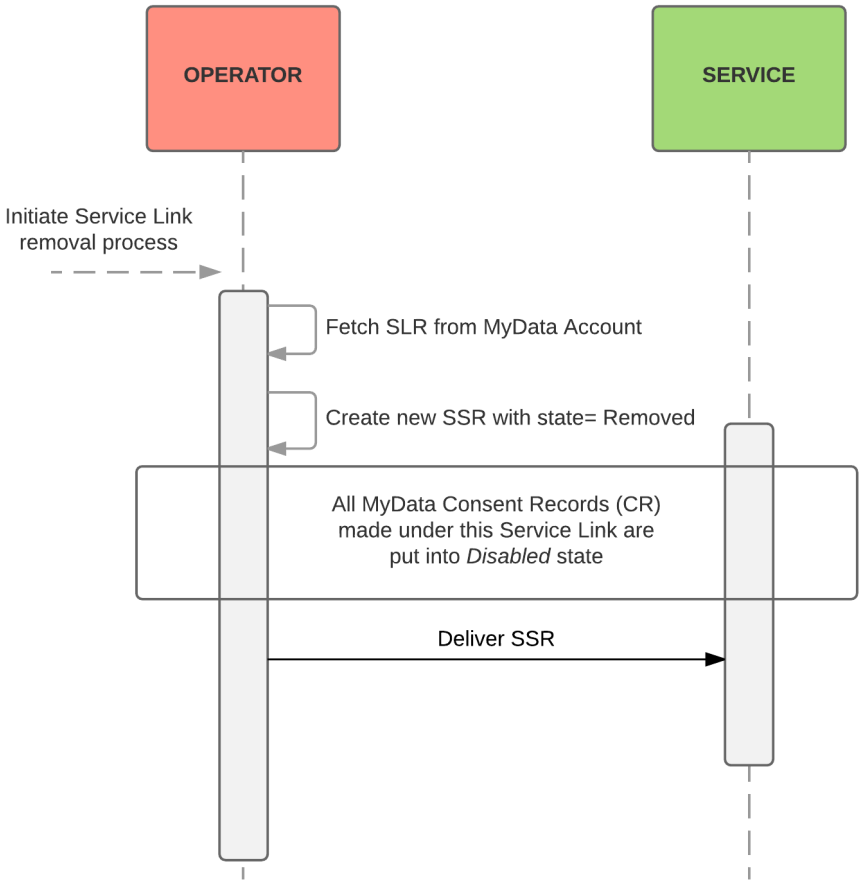

Figure 3.3: Description of Service Link removal process

3.3 Request a New Copy of SLR from Operator

Service MUST request a new copy of SLR e.g. in case service's copy of SLR is lost. Service MAY request a new copy of SLR even if it has an existing copy. New copy of SLR is requested using the Surrogate ID.

Prerequisites: Service has been linked to MyData Account
Process: Service requests a new copy of SLR
Outcome: Service receives a new copy of SLR.

A simplified flow is shown in Figure 3.4.

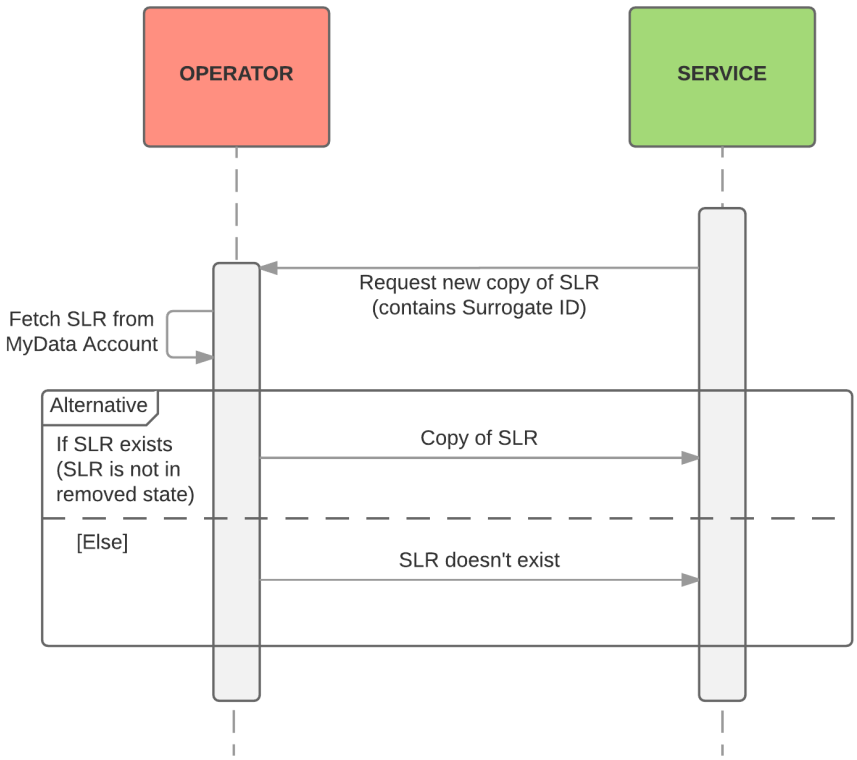

Figure 3.4: Flow of requesting the new SLR copy

3.4 Request a New Copy of SSR from Operator

Service MUST request a new copy of Service Link Status Record (SSR) e.g. in case service's copy of the record is lost. Service MAY request a new copy of SSR even if it has an existing copy. New copy of SSR is requested using the Surrogate ID.

Prerequisites: Service has been linked to MyData Account.
Process: Service requests a new copy of SSR.
Outcome: Service receives a new copy of SSR.

A simplified flow is shown in Figure 3.5.

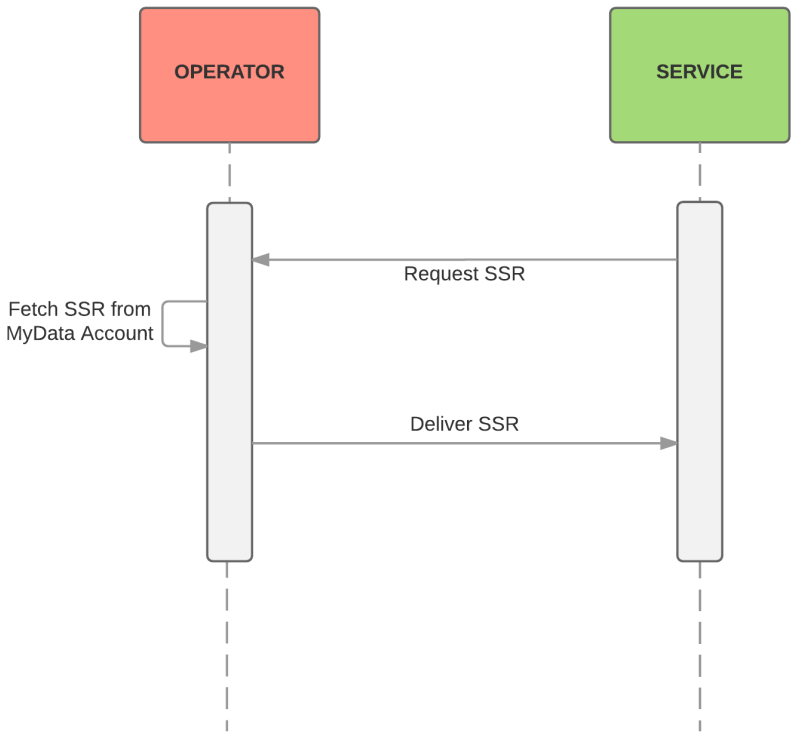

Figure 3.5: Flow of requesting the new Service Link Status Record

4. Service Link Record

A Service Link consists of two records: MyData Service Link Record (SLR) and Service Link Status Record (SSR).

4.1 Structure of Service Link Record

Table 4.1. Key-value pairs of Service Link Record, their description and entities responsible for the needed information.

KEY	TYPE	DESCRIPTION	ADDITIONAL DETAILS
version	String	Version number of Service Link Record specification. For this release MUST be 1.2.1	
link_id	String	Unique ID of this Service Link Record. Unique within the Operator where this Service Link Record was constructed.	
operator_id	String	Unique ID of the Operator	
service_id	String	ID of the service which this linking has been / is being made. Unique within the Service Registry where this Service is registered.	Service Registry generates an ID for this Service when it is registered to the Service Registry.
surrogate_id	String	ID representing Account Owner in communication between Operator and service. Unique within this Service Link.	Service generates surrogate_id during Service Linking process.
operator_key	JWK	Operator's public key used to verify operator issued status change messages. JWK structure MUST contain 'kid' parameter.	
cr_keys	JSON Web Key (JWK) Set structure	Account Owner's public key(s) [1..*] used to verify MyData Consent and Consent Status Records delivered to Service. Each JWK structure MUST contain 'kid' parameter.	
iat	Integer	Timestamp (UTC) when this Service Link Record was constructed.	

Service Link Record MUST be signed with the account owner's and Service's private keys as defined in [RFC7515]. The JWS headers MUST contain 'kid' fields identifying account owner's and service's key pairs used to sign Service Link Record.

4.1.1 Service Link Record payload

```
{
  "version": "String",
  "link_id": "String",
  "operator_id": "String",
  "service_id": "String",
  "surrogate_id": "String",
  "iat": "Integer",
  "operator_key": {
    "jwk": "JSON Web Key (JWK) presentation of Operator's public key used to verify
operator issued status change messages"
  },
  "cr_keys": {
    "keys": [
      "JSON Web Key (JWK) presentation of public part of key used for signing consents"
    ]
  }
}
```

4.2 Structure of Service Link Status Record

Table 4.3 presents the structure of the MyData Service Link Status Record.

Table 4.3 Service Link Status Record

KEY	TYPE	DESCRIPTION
version	String	Version number of Service Link Status Record specification. For this release MUST be 1.2.1
record_id	String	Unique ID of the record
surrogate_id	String	The Account owner's Surrogate ID
slr_id	String	Unique ID of the Service Link
sl_status	String	Active/Removed
iat	Integer	Time when the Status Record was issued
prev_record_id	String	Link to previous Status Record ID, NULL if first Status Record

Service Link Status Record MUST be signed with the account owner's private keys as defined in [RFC7515].

The JWS header MUST contain ‘kid’ field identifying account owner’s key pair used to sign the Service Link Status Record.

4.2.1 Service Link Status Record payload

```
{
  "version": "String",
  "record_id": String,
  "surrogate_id": String,
  "slr_id": String,
  "sl_status": String,
  "iat": Integer,
  "prev_record_id": String
}
```

5. Service Linking APIs

Service Linking has been implemented as a service in MyData SDK. Related developer API documentation and detailed flow diagrams clarifying the implementation are linked and documented in this section.

5.1. Interfaces of different actors

There are two main interfaces: Service Link Management API (provided by the Operator) and Service Linking API (to be provided by Source and Sink each).

5.2 API Specification

Operator’s Service Link Management interface:
http://editor.swagger.io/#/?import=https://raw.githubusercontent.com/HIIT/mydata-sdk/v1.2.1/Operator_Components/doc/api/swagger_Operator_SLR.yml

Service Linking API for the corresponding endpoint in Sources and Sinks, shown below in 5.2 flow figures as ‘Service Management (Service)’:
http://editor.swagger.io/#/?import=https://raw.githubusercontent.com/HIIT/mydata-sdk/v1.2.1/Service_Components/doc/api/swagger_Service_Mgmnt.yml

5.3. Detailed Flow

Below two figures show details of the service link flows as currently implemented in MyData SDK. Figure 5.1 covers the optional phase required to obtain a Surrogate ID, and Figure 5.2 covers the actual service linking flow that obtains SLR and SSR.

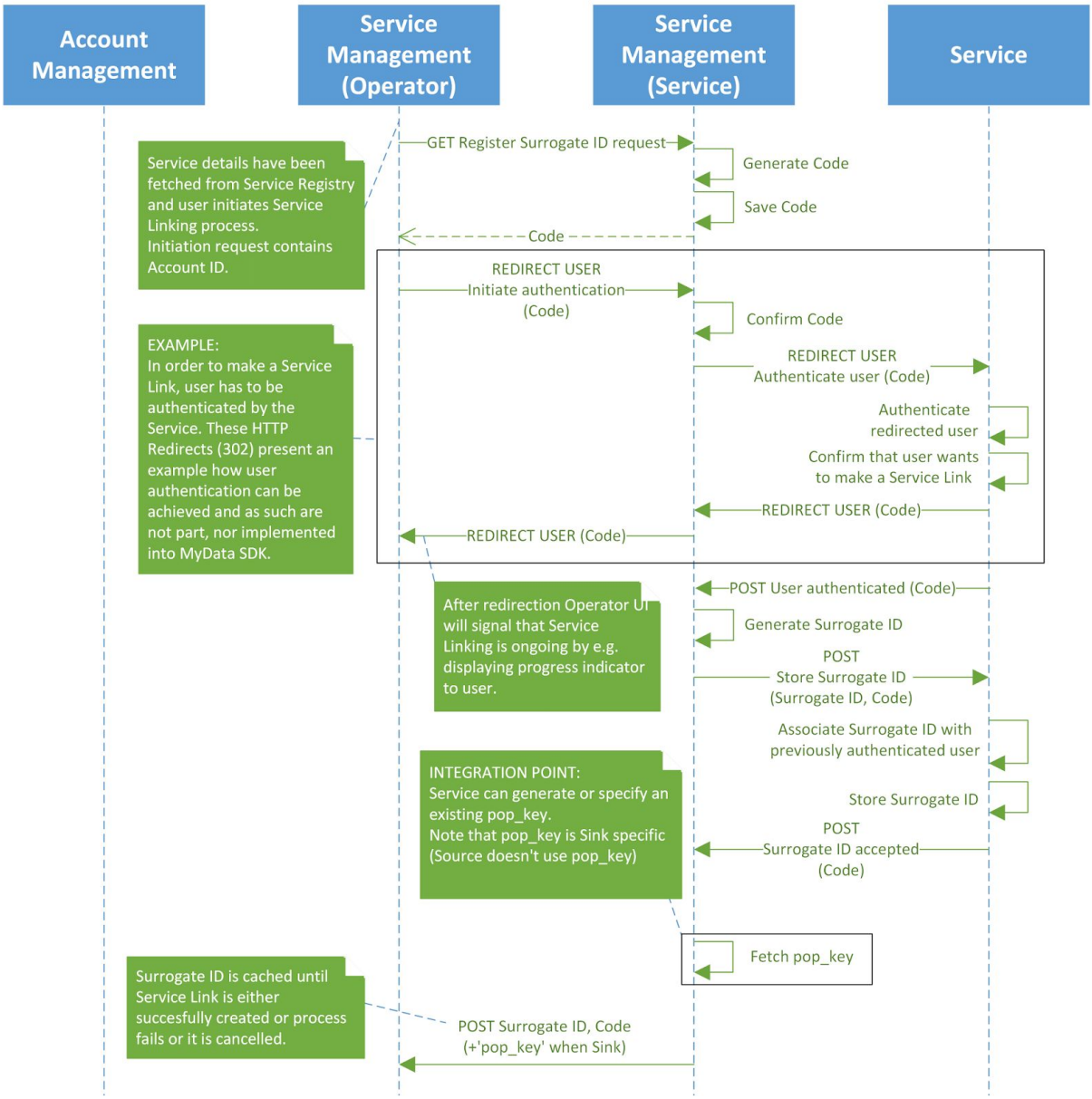

Figure 5.1: Getting Surrogate ID

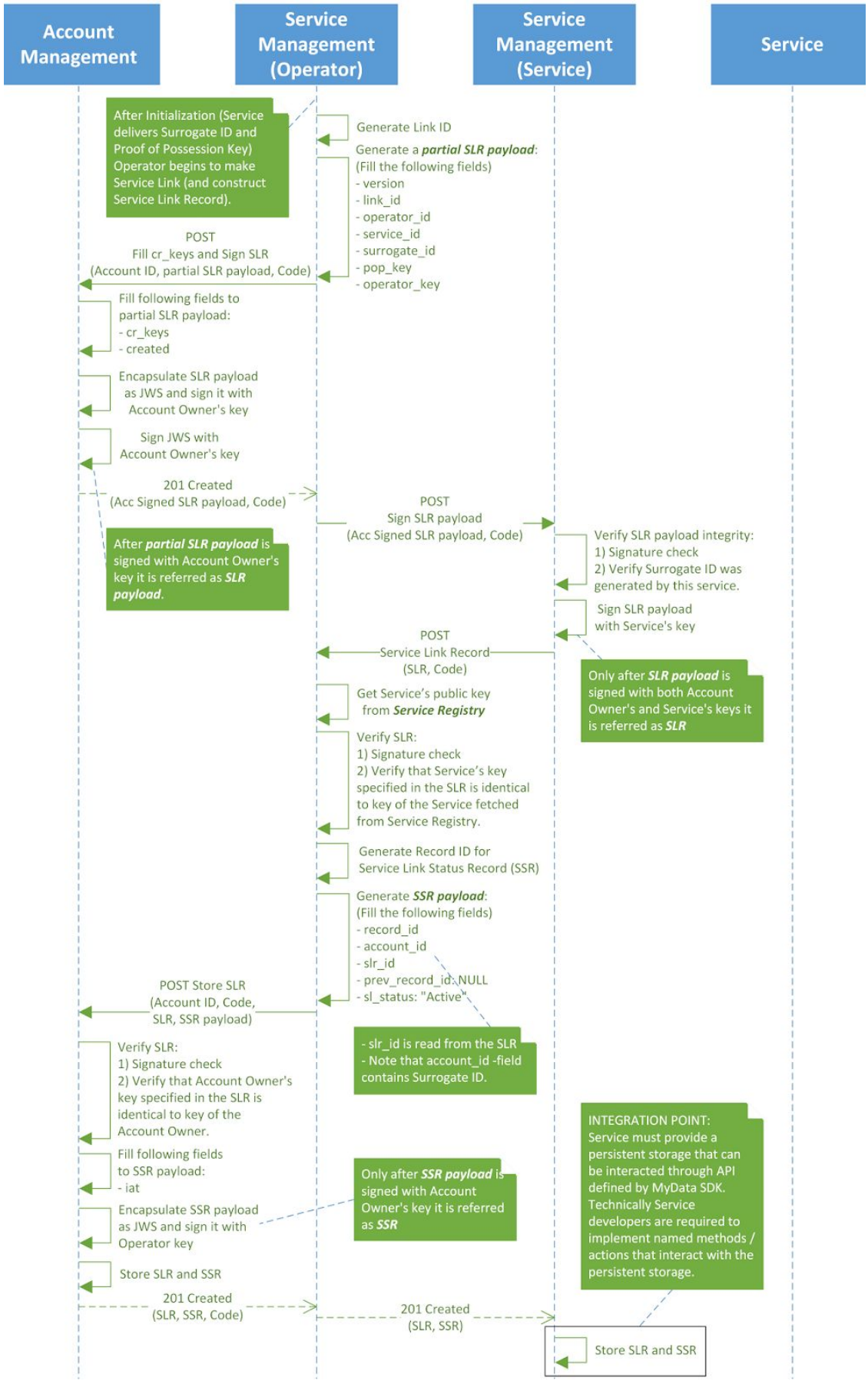

Figure 5.2: Creating the SLR and SSR

References

[RFC2119] Bradner, S, "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
[RFC7515] Jones, M, Bradley, J, Sakimura, N, JSON Web Signature”, RFC 7515, May 2015.
[RFC7519] Jones, M., Bradley, J., Sakimura, N. “JSON Web Token (JWT)”, RFC 7519, May 2015