Service Design for S-421 Route Plan using SECOM

Contents

[1 Introduction 6](#_Toc158623609)

[1.1 Purpose of the Document 6](#_Toc158623610)

[1.2 Intended Readership 6](#_Toc158623611)

[1.3 Inputs from Other Sources 6](#_Toc158623612)

[2 Service Identification 7](#_Toc158623613)

[3 Technology Introduction 8](#_Toc158623614)

[3.1 General 8](#_Toc158623615)

[3.2 Service Technology and transportation protocol 8](#_Toc158623616)

[3.3 Security 8](#_Toc158623617)

[3.3.1 Communication channel security 8](#_Toc158623618)

[3.3.2 Data Protection 8](#_Toc158623619)

[3.3.3 Data Signature 9](#_Toc158623620)

[3.3.4 Data Encryption 9](#_Toc158623621)

[4 Service Design Overview 10](#_Toc158623622)

[4.1 General 10](#_Toc158623623)

[4.2 Service interfaces 10](#_Toc158623624)

[5 Physical Data Model 13](#_Toc158623625)

[5.1 Payload Data Model 13](#_Toc158623626)

[6 Service Interface Design 14](#_Toc158623627)

[6.1 General 14](#_Toc158623628)

[6.2 Operation UPLOAD 14](#_Toc158623629)

[6.2.1 Operation Functionality 14](#_Toc158623630)

[6.2.2 Operation Parameters 14](#_Toc158623631)

[6.2.3 Values for S-421 Route Exchange Service 15](#_Toc158623632)

[6.3 Operation GET 16](#_Toc158623633)

[6.3.1 Operation Functionality 16](#_Toc158623634)

[6.3.2 Operation Parameters 16](#_Toc158623635)

[6.3.3 Values for S-421 Route Exchange Service 17](#_Toc158623636)

[6.4 Operation UPLOAD LINK 18](#_Toc158623637)

[6.5 Operation GET BY LINK 18](#_Toc158623638)

[6.6 Operation ACKNOWLEDGEMENT 19](#_Toc158623639)

[6.6.1 Operation Functionality 19](#_Toc158623640)

[6.6.2 Operation Parameters 19](#_Toc158623641)

[6.6.3 Values for S-421 Route Exchange Service 20](#_Toc158623642)

[6.7 Operation GET SUMMARY 21](#_Toc158623643)

[6.7.1 Operation Functionality 21](#_Toc158623644)

[6.7.2 Operation Parameters 21](#_Toc158623645)

[6.7.3 Values for S-421 Route Exchange Service 22](#_Toc158623646)

[6.8 Operation ACCESS 23](#_Toc158623647)

[6.8.1 Operation Functionality 23](#_Toc158623648)

[6.8.2 Operation Parameters 23](#_Toc158623649)

[6.8.3 Values for S-421 Route Exchange Service 23](#_Toc158623650)

[6.9 Operation ACCESS NOTIFICATION 23](#_Toc158623651)

[6.9.1 Operation Functionality 24](#_Toc158623652)

[6.9.2 Operation Parameters 24](#_Toc158623653)

[6.9.3 Values for S-421 Route Exchange Service 24](#_Toc158623654)

[6.10 Operation SUBSCRIPTION 25](#_Toc158623655)

[6.10.1 Operation Functionality 25](#_Toc158623656)

[6.10.2 Operation Parameters 25](#_Toc158623657)

[6.10.3 Values for S-421 Route Exchange Service 25](#_Toc158623658)

[6.11 Operation SUBSCRIPTION NOTIFICATION 27](#_Toc158623659)

[6.11.1 Operation Functionality 27](#_Toc158623660)

[6.11.2 Operation Parameters 27](#_Toc158623661)

[6.11.3 Values for S-421 Route Exchange Service 27](#_Toc158623662)

[6.12 Operation REMOVE SUBSCRIPTION 28](#_Toc158623663)

[6.12.1 Operation Functionality 28](#_Toc158623664)

[6.12.2 Operation Parameters 28](#_Toc158623665)

[6.12.3 Values for S-421 Route Exchange Service 28](#_Toc158623666)

[6.13 Operation CAPABILITY 29](#_Toc158623667)

[6.13.1 Operation Functionality 29](#_Toc158623668)

[6.13.2 Operation Parameters 29](#_Toc158623669)

[6.13.3 Values for S-421 Route Exchange Service 30](#_Toc158623670)

[6.14 Operation PING 31](#_Toc158623671)

[6.14.1 Operation Functionality 31](#_Toc158623672)

[6.14.2 Operation Parameters 31](#_Toc158623673)

[6.14.3 Values for S-421 Route Exchange Service 31](#_Toc158623674)

[6.15 Operation ENCRYPTIONKEY 32](#_Toc158623675)

[6.15.1 Operation Functionality 32](#_Toc158623676)

[6.15.2 Operation Parameters 32](#_Toc158623677)

[6.15.3 Values for S-421 Route Exchange Service 33](#_Toc158623678)

[6.16 Operation ENCRYPTIONKEY NOTIFICATION 34](#_Toc158623679)

[6.16.1 Operation Functionality 34](#_Toc158623680)

[6.16.2 Operation Parameters 34](#_Toc158623681)

[6.16.3 Values for S-421 Route Exchange Service 34](#_Toc158623682)

[6.17 Operation GET PUBLICKEY 35](#_Toc158623683)

[6.17.1 Operation Functionality 35](#_Toc158623684)

[6.17.2 Operation Parameters 35](#_Toc158623685)

[6.17.3 Values for S-421 Route Exchange Service 35](#_Toc158623686)

[6.18 Operation UPLOAD PUBLICKEY 36](#_Toc158623687)

[6.18.1 Operation Functionality 36](#_Toc158623688)

[6.18.2 Operation Parameters 36](#_Toc158623689)

[6.18.3 Values for S-421 Route Exchange Service 36](#_Toc158623690)

[7 Service Dynamic Behaviour 37](#_Toc158623691)

[8 References 38](#_Toc158623692)

[9 Acronyms and Terminology 39](#_Toc158623693)

[1.4 Acronyms 39](#_Toc158623694)

[1.5 Terminology 39](#_Toc158623695)

Table of figures

**No table of figures entries found.**

List of tables

N/A

# Introduction

This document was produced as part of the work of IALA joint VTS-ENAV task group on development of technical service specifications for VTS. The document is structured according to the IALA Guideline *G1128 The Specification of e-Navigation Technical Services* [1].

This service design is intended to support any service specification that involve exchange of S-421 Route Plan. The service design can be part of a service orchestrated architecture.

This service design refers to the generic SECOM Service Design Template.

## Purpose of the Document

The purpose of this service specification is to provide a design for the implementation of Route Plan exchange using SECOM and S-421 as the S-100 series data model for the actual message payload.

## Intended Readership

This service design is intended to be read by service architects, system engineers and developers in charge of designing and developing an instance exchanging S-421 Route Plan using SECOM.

Furthermore, this service specification is intended to be read by enterprise architects, service architects, information architects, system engineers and developers in pursuing architecting, designing and development activities of other related services.

## Inputs from Other Sources

As this design uses SECOM and an understanding of IEC 63173-2 SECOM is recommended.

This design is based on Service Design – Template SECOM REST and uses text from the template where valid.

# Service Identification

The purpose of this chapter is to provide a unique identification of the service and describe where the service is in terms of the engineering lifecycle.

|  |  |
| --- | --- |
| Name | Route Exchange REST Service Design using SECOM |
| Implements | TBD |
| ID | urn:mrn:iala:techsvc:design:vts:secom:routeplan |
| Version | 0.1 |
| Description | The S-421 Route Plan Service Design using SECOM specifies how S-421 Route Plan service is implemented using SECOM to facilitate the exchange between ship and shore systems. |
| Keywords | VTS, MS1, Route Plan, Ship Traffic Management, S-421, SECOM |
| Architect(s) | IALA TG 2 |
| Status | provisional |

# Technology Introduction

## General

This design realizes the service specification **Error! Reference source not found.** using SECOM as defined in **Error! Reference source not found.**.

The services conforming to this design must be implemented with REST APIs using HTTPS with TLS protection to encrypt all communication in transit.

## Service Technology and transportation protocol

**Reference:** IEC 63173-2 SECOM v1.0.0 Clause 5.3 Service Technology

The technology (architectural style) chosen is REST (REpresentational State Transfer) upon HTTP/1.1 (RFC 7231).

## Security

### Communication channel security

**Reference:** IEC 63173-2 SECOM v1.0.0 Clause 6 SECOM communication channel security

The channel security between the service and a consumer are:

• HTTP/1.1 according to RFC-7231

• HTTPS over TLS according to RFC-2818

Valid versions of TLS for this version of service design template are:

• TLS version 1.1 (RFC-4346)

• TLS version 1.2 (RFC-5246)

• TLS version 1.3 (RFC-8446)

X.509 Certificates are used in the TLS according to RFC 5280 and RFC 2459.

Certificates shall be verified with OCSP and/or CRL methods.

### Data Protection

**Reference:** IEC 63173-2 SECOM v1.0.0 Clause 7 SECOM data protection

**Reference:** IHO Standard S-100 ed5.0.0 Part 15 Data Protection Scheme

The data is mandatory to be signed by the sender to enable data authentication and integrity check by the receiver.

The data can optionally be encrypted by the sender, and the sender is responsible for exchanging the encryption key to receiver.

The data (one or more data files) can optionally be packaged and compressed before signed

### Data Signature

**Reference:** IEC 63173-2 SECOM v1.0.0 Clause 7.3 Data authentication and signing

**Reference:** IHO Standard S-100 ed5.2.0 Part 15-8 Data Authentication

**Reference:** NIST Digital Signature Standard (DSS–FIPS Publication 186)

The algorithm for signing data is ECDSA and SHA256.

The signature is transported in HEX.

### Data Encryption

**Reference:** IEC 63173-2 SECOM v1.0.0 Clause 7.4 Data encryption

**Reference:** IHO Standard S-100 ed5.0.0 Part 15-6 Data Encryption

The encryption algorithm for encryption is AES (128, 192 or 256 bit) and CBC mode.

The symmetric encryption key can be exchanged by different means, including using the SECOM REST API and Diffie Helman to encrypt and exchange the encryption key.

# Service Design Overview

## General

This service design is based on SECOM Service Design Template **Error! Reference source not found.** and from that the service interfaces relevant for S-421 have been chosen. This service design is intended to be used by any actor, independent on purpose of exchanging the S-421 Route Plan, hence all interfaces from the SECOM Service Design Template are described here.

## Service interfaces

This chapter is based on the description in ref **Error! Reference source not found.** and ref **Error! Reference source not found.** and only describes the additional information required to tailor the service interface to S-421 Route Plan data.

|  |  |  |
| --- | --- | --- |
| Interface | SECOM Reference | Comment |
| Capability | IEC 63173-2 SECOM v1.0.0 Clause 5.7.13 service interface – Capability | This interface is called when client asks for the service capabilities. Required by SECOM standard. |
| Ping | IEC 63173-2 SECOM v1.0.0 Clause 5.7.14 service interface – Ping | This interface is called when client checks the availability of the service. Required by SECOM standard. |
| Upload | IEC 63173-2 SECOM v1.0.0 Clause 5.7.2 service interface – Upload | This interface is called when client uploads (pushes) S-421 Route Plan to the service. |
| Acknowledgement | IEC 63173-2 SECOM v1.0.0 Clause 5.7.4 service interface – Acknowledgement | Interface to send acknowledgement on uploaded information. |
| Get | IEC 63173-2 SECOM v1.0.0 Clause 5.7.5 service interface – Get | Interface to ask for (pull) S-421 Route Plans from provider. |
| Get Summary | IEC 63173-2 SECOM v1.0.0 Clause 5.7.6 service interface – Get Summary | Interface to ask for (pull) a list of available S-421 Route Plans from provider |
| Subscription | IEC 63173-2 SECOM v1.0.0 Clause 5.7.10 service interface – Subscription | Interface to create subscription of S-421 Route Plans. |
| Remove Subscription | IEC 63173-2 SECOM v1.0.0 Clause 5.7.11 service interface – Remove Subscription | Interface to remove subscription on S-421 Route Plans. |
| Subscription Notification | IEC 63173-2 SECOM v1.0.0 Clause 5.7.12 service interface – Subscription Notification | Interface for notification from subscription events created by the provider. |
| Access | IEC 63173-2 SECOM v1.0.0 Clause 5.7.8 service interface – Access | Interface to ask for access to S-421 Route Plans |
| Access Notification | IEC 63173-2 SECOM v1.0.0 Clause 5.7.9 service interface – Access Notification | Interface for notification from access request |
| Encryption Key | IEC 63173-2 SECOM v1.0.0 Clause 5.7.15 service interface – EncryptionKey | Interface to securely send symmetric key for data encryption |
| Public Key | IEC 63173-2 SECOM v1.0.0 Clause 5.7.16 service interface – PublicKey | Interface to request (pull) and send (push) a public certificate |

There are three components that are of interest from the perspective of the service design:

* The service has a SECOM-component which supports the SECOM REST APIs defined in the table above. All other components of the service are left to the decisions of the implementing party.
* The vessel has a SECOM-component which will accepts the incoming connections from the service and store all messages until delivered to the vessel. This interface is typically on a shoreside server as it must be always available and at a static address.
* The vessel has an implementation of a SECOM client which allows it to make direct SECOM calls to the service without having to proxy all calls via the SECOM-component on shore.

In this service design we will not define the communication between the service and shore system or the ship and the ship’s registered SECOM interface. These are specific for each implementation and depend on the shore system and ship’s system.

# Physical Data Model

The data model of the service is a combination of JSON (SECOM calls) and XML (the S-421 payload). The SECOM JSON is defined in [7] section 5.

## Payload Data Model

**[Source IEC 63173-1 S-421]**

Refer to URL for XSD file to S-421

Refer to IEC 63173-1 Route Plan

Diagram

Description automatically generated

1. Payload Data Model diagram

# Service Interface Design

## General

This section describes the details of each service interface and its operations.

The Service Interface design covers the static design description while the dynamic design (behaviour) is described in section 7 Service Dynamic Behaviour.

The following definitions of the interface and operations are based on SECOM Service Design Template, see ref **Error! Reference source not found.**.

Each operation has been added with a section for the specific S-421 Route Plan service details as a complement to the design template information.

## Operation UPLOAD

This interface is called when the client uploads (pushes) data to the service. The sender (client) decides on the format and protection of the data. If acknowledgement is requested, it will be given by callback to interface Acknowledgement.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/object {body} : response**

The data (payload) package with its metadata.

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Format | Require |
| envelope  data | string | Base64 | Required |
| envelope  containerType | integer | enum | Required |
| envelope  dataProductType | integer | enum | Required |
| envelope  exchangeMetadata  dataProtection | boolean |  | Required |
| envelope  exchangeMetadata  protectionScheme | string |  | Required |
| envelope  exchangeMetadata  dataSignatureReference | string |  | Required |
| envelope  exchangeMetadata  dataSignatureValue  publicRootCertificateThumbprint | string | SHA1 | Optional |
| envelope  exchangeMetadata  dataSignatureValue  publicCertificate | string | Minified PEM | Required |
| envelope  exchangeMetadata  compressionFlag | boolean |  | Required |
| envelope  fromSubscription | boolean |  | Optional |
| envelope  ackRequest | boolean |  | Required |
| envelope  transactionIdentifier | string | uuid | Required |
| envelope  envelopeSignatureCertificate | string | Minified PEM | Required |
| envelope  envelopeRootCertificateThumbprint | string | SHA1 | Optional |
| envelope  envelopeSignatureTime | string | date-time | Optional |
| envelopeSignature | string | HEX | Required |

Table 1 Service body

|  |  |  |  |
| --- | --- | --- | --- |
| Response Code | Object | Type | Description |
| 200 | message | string | Message successfully uploaded |
| Secom\_ResponseCodeEnum | integer |  |
| 400 | message | string | Bad request |
| 401 | message | string | Unauthorized |
| 403 | message | string | Not authorized to upload link |
| 405 | message | string | Method not allowed |
| 500 | message | string | Internal server error |
| 501 | message | string | Not implemented |

Table 2 Service response

### Values for S-421 Route Exchange Service

The default value for data is one base64 encoded S-421 XML message.

The default value for containerType is S100\_DataSet.

The dataProductType is 24/25 for S421 Route plan.

The default value for dataProtection is false, the S-421 Route Plan is not encrypted, but it can optionally be set to true if encrypted. The encryptionKey must then also be exchanged.

The default value for protectionScheme is “SECOM”.

The dataSignatureReference is “ECDSA”.

The default value for compressionFlag is false, but optionally it can be set to true if the data is instead a base64 encoded ZIP compressed S421 XML message.

If the S-421 Route Plan has been uploaded within a subscription, the subscriptionFlag is set to true.

## Operation GET

This interface is called when the client gets (pulls) data from the service.

The Get interface is used for pulling information from a service provider. The owner of the information (provider) is responsible for the authorization procedure before returning information. The consumer can ask for information by its reference, geometry, time or arbitrary query for e.g. status on the information product. If no filtering parameters are given, all authorized information is to be sent. The information owner decides what information the consumer is authorized to based on the identity in the TLS client certificate i.e. the identity the service instance belongs to.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**GET baseUrl/v1/object/pathParams?queryParams : response**

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Format | Require |
| dataReference | string | uuid | optional |
| containerType | integer | enum | optional |
| dataProductType | integer | enum | optional |
| productVersion | string |  | optional |
| geometry | string | WKT | optional |
| unlocode | string | [A-Z]{5} | optional |
| validFrom | string | date-time | optional |
| validTo | string | date-time | optional |
| page | integer |  | optional |
| pageSize | integer |  | optional |

Table 3 Service parameter

|  |  |  |  |
| --- | --- | --- | --- |
| HTTP Code | Content | Type | Description |
| 200 | application/json | string | The operation was performed successfully |
| array of { |  |  |
| data | string |  |
| exchangeMetadata  dataProtection | boolean | True if data encrypted |
| exchangeMetadata  protectionScheme | string | Reference to protection schema administrator |
| exchangeMetadata  digitalSignatureReference | string |  |
| exchangeMetadata  digitalSignatureValue  publicRootCertificateThumbprint | string |  |
| exchangeMetadata  digitalSignatureValue  publicCertificate | string |  |
| exchangeMetadata  digitalSignatureValue  digitalSignature | string |  |
| exchangeMetadata   compressionFlag | boolean | True if compressed |
| ackRequest | integer |  |
| } |  |  |
| pagination  totalItems | integer |  |
| pagination  maxItemsPerPage | integer |  |
| 400 | application/json | string | Bad request |
| 401 | application/json | string | Unauthorized |
| 403 | application/json | string | Not authorized to requested information |
| 404 | application/json | string | Information not found |
| 405 | application/json | string | Method not allowed |
| 500 | application/json | string | Internal server error |
| 501 | application/json | string | Not implemented |

Table 4 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation UPLOAD LINK

The purpose with this interface is to upload (push) a link to information to a consumer. Hence, a consumer shall implement this interface in order to receive a link to the information that can be retrieved. This interface shall be used when large amounts of data shall be exchanged. The provider of information then uploads a link to a consumer, and the consumer then uses the Get by Link interface to pull the data from the provider.

This operation is not foreseen to be useful for exchange of S-421 Route Plans.

## Operation GET BY LINK

This interface is called when the client downloads (pulls) large (amounts of?) data by reference given from interface Upload Link.

The Get By Link interface is used for pulling information from a data storage handled by the information owner. The link to the data storage can be exchanged with Upload Link interface. The owner of the information (provider) is responsible for the relevant authentication and authorization procedures before returning information.

This operation is not foreseen to be useful to exchange S-421 Route Plans.

## Operation ACKNOWLEDGEMENT

This interface is called as response to Acknowledgement request in Upload.

During upload of information, an acknowledgement can be requested which is expected to be asynchronously received when the uploaded message has been delivered to the end system (technical acknowledgement), and an acknowledgement when the message has been opened and/or processed by the end user (operational acknowledgement). The acknowledgement contains a reference to the object delivered and has no time limit.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/acknowledgement {body} : response**

Object with reference to information and time when delivered

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| envelope  createdAt | string | date-time | Required |
| Envelope  envelopeCertificate | string | Minified PEM | Required |
| Envelope envelopeRootCertificateThumbprint | string | SHA1 | Required |
| Envelope  transactionIdentifier | string | uuid | required |
| Envelope  ackType | integer | enum | required |
| Envelope  nackType | integer | enum | optional |
| Envelope  envelopeSignatureTime | string | date-time | required |
| digitalSignature | string | HEX | required |

Table 5 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json | string |  | The operation was performed successfully |
| message | string |  |  |
| SECOM\_code | integer | enum |  |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 403 | application/json | string |  | Not authorized to upload ACK |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 6 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation GET SUMMARY

This interface is called when the client wants a summary of available data from the service. The actual data is retrieved (pulled) using the interface Get.

A list of information shall be returned from this interface. The summary contains identity, status, size and a short description of each information object. The actual information object shall be retrieved using the Get interface. The consumer can ask for information by geometry, location and time. If no filtering parameters are given, available summary information is to be sent.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**GET baseUrl/v1/object/summary/pathParam?queryParam : response**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Param | Param | Type | Format | Require |
| containerType | query | integer | enum | optional |
| dataProductType | query | integer | enum | optional |
| productVersion | query | string |  | optional |
| geometry | query | string | WKT | optional |
| unlocode | query | string | 5[A-Z] | optional |
| validFrom | query | string | date-time | optional |
| validTo | query | string | date-time | optional |
| page | query | integer |  | optional |
| pageSize | query | integer |  | optional |

Table 7 Service parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json | string |  | The operation was performed successfully |
| array of { | | | |
| informationSummaryObject  dataReference | string |  |  |
| informationSummaryObject  dataProtection |  |  |  |
| informationSummaryObject  dataCompression |  |  |  |
| informationSummaryObject  containerType | integer | enum |  |
| informationSummaryObject  dataProductType | integer | enum |  |
| informationSummaryObject  info\_productVersion | string |  |  |
| informationSummaryObject  info\_identifier | string |  |  |
| informationSummaryObject  info\_name | string |  |  |
| informationSummaryObject  info\_Status | string |  |  |
| informationSummaryObject  info\_Description | string |  |  |
| informationSummaryObject  info\_lastModifiedDate | string | date-time |  |
| informationSummaryObject  info\_size | string | integer |  |
| } | | | |
| pagination  totalItems | integer |  |  |
| pagination  maxItemsPerPage | integer |  |  |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 8 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation ACCESS

This interface is called when the client asks for access to data from the service. Response is given by callback to interface Access Notification.

Access to information can be requested through the Access interface. The result is sent asynchronously through the Access Notification interface.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/access {body} : response**

Description of reason for requesting access to information

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| reason | string |  | required |
| reasonEnum | integer | enum | required |
| containerType | integer | enum | optional |
| dataProductType |  |  | optional |
| dataReference | string |  | optional |
| productVersion | string |  | optional |

Table 9 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json | string |  | The operation was performed successfully |
|  | integer |  |  |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 10 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation ACCESS NOTIFICATION

This interface is called as callback response to interface Access.

The result from Access Request shall be sent asynchronously through this interface.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/access/** **notification {body} : response**

Result from the request for access; True or False

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| decision | boolean |  | required |
| decisonReason | string |  | required |
| transactionIdentifier | string | uuid | required |

Table 11 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json | string |  | The operation was performed successfully |
| SECOM\_code | integer | enum |  |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 12 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation SUBSCRIPTION

This interface is called when the client or server initiates subscription on data from the service. Response is given as callback to interface Upload and Subscription Notification.

The purpose of the interface is to request subscription on information, either specific information according to parameters, or the information accessible upon decision by the information provider. Each subscription request reflects one parameter query set.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/** **subscription {body} : response**

Specific id on the information object subscription is requested for specific status on the information or specific area of interest

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| containerType | integer | enum | optional |
| geometry | string |  | optional |
| unlocode | string |  | optional |
| dataProductType | integer | enum | optional |
| productVersion | string |  | optional |
| dataReference | string |  | optional |
| subscriptionPeriodStart | string | date-time | optional |
| subscriptionPeriodEnd | string | date-time | optional |

Table 13 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json | string |  | The operation was performed successfully |
| message | string |  |  |
| subscriptionIdentifier | string | uuid |  |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 14 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation SUBSCRIPTION NOTIFICATION

This interface is called as callback response from interface Subscription or Remove Subscription.

The interface receives notifications when subscription is created or removed by information producer.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/subscription/notification {body} : response**

Contains the identity of the information object in focus and type of event; Create or Delete.

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| subscriptionIdentifier | string | uuid | required |
| eventEnum | integer | enum | required |

Table 15 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json | string |  | The operation was performed successfully |
| message | string |  |  |
| subscriptionIdentifier | string | uuid |  |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 16 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation REMOVE SUBSCRIPTION

This interface is called when the client or server removes a subscription. Response is given as callback to interface Subscription Notification.

Subscription(s) can be removed either internally by the information owner, or externally by the consumer. This interface shall be used by the consumer to request removal of subscription.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**DELETE baseUrl/v1/** **subscription {body} : response**

Specific identity of the information object to remove subscription for. If no id entity provided, all subscriptions for the caller is removed.

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| subscriptionIdentifier | string | uuid | required |

Table 17 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json | string |  | The operation was performed successfully |
| message | string |  | Subscription \*identifier\* removed |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 403 | application/json | string |  | Not authorized to remove subscription |
| 404 | application/json | string |  | Subscription \*identifier\* not found |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 18 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation CAPABILITY

This interface is called when the client asks for the service capabilities.

The purpose of the interface is to provide a dynamic method to ask a service instance at runtime what interfaces are accessible, and what payload formats and versiosn that are valid.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**GET baseUrl/v1/capability/pathParam?queryParam : response**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json |  |  | The operation was performed successfully |
| array of CapabilityObject { | | | |
| dataProductType | integer | enum |  |
| containerType | integer | enum |  |
| productSchemaUrl | string |  |  |
| serviceVersion | string |  |  |
| implementedInterfaces  upload | boolean |  |  |
| implementedInterfaces  uploadLink | boolean |  |  |
| implementedInterfaces  get | boolean |  |  |
| implementedInterfaces  getByLink | boolean |  |  |
| implementedInterfaces  getSummary | boolean |  |  |
| implementedInterfaces  subscription | boolean |  |  |
| implementedInterfaces  access | boolean |  |  |
| implementedInterfaces  encryptionKey | boolean |  |  |
| } | | | |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 19 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation PING

This interface is called when the client checks the availability of the service.

The purpose of the interface is to provide a dynamic method to ask for the technical status of the specific service instance.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**GET baseUrl/v1/**ping**/ pathParam?queryParam : response**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json |  |  | The operation was performed successfully |
| message | string |  | Response message |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 20 Service Response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation ENCRYPTIONKEY

This interface is called when sending (pushing) an encryption key to a receiver.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/encryptionKey {body} : response**

Object with symmetric encryption key

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| envelope  encryptionKey | string | byte | required |
| envelope  iv |  |  | required |
| envelope  transactionIdentifier | string | uuid | optional |
| envelope  digitalSignatureValue  publicRootCertificateThumbprint | string | SHA1 | optional |
| envelope  digitalSignatureValue  publicCertificate | string | PEM | required |
| envelope  digitalSignatureValue  digitalSignature | string | HEX | required |
| envelope  envelopeSignatureCertificate | string | PEM | required |
| envelope  envelopeRootCertificateThumbprint | string | SHA1 | required |
| envelope  envelopeSignatureTime | string | date-time | required |
| envelopeSignature | string | HEX | required |

Table 21 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json |  |  | The operation was performed successfully |
| message | string |  | Response message |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 22 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation ENCRYPTIONKEY NOTIFICATION

This interface is called when sending (pushing) en encryption key to a receiver.

The purpose of this interface is to receive a request for an exchange of an encrypted secret key. The response is sent asynchronously through the consumer’s POST encryption key operation.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/encryptionKey/notify {body} : response**

EncryptionKey notification object

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Body | Type | Format | | Require |
| envelope  dataReference | string | | uuid | required |
| envelope  publicCertificate | string | | PEM | required |
| envelope  envelopeSignatureCertificate | string | | PEM | required |
| envelope  envelopeSignatureTime | string | | date-time | required |
| envelopeSignature | string | | HEX | required |

Table 23 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 202 | application/json |  |  | Request accepted |
| message | string |  | Response message |
| SECOM\_Code | integer | enum |  |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 24 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation GET PUBLICKEY

This interface is called when the client gets (pulls) the public certificate(s) from the service.

The purpose of the interface is to request a public key.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**GET baseUrl/v1/PublicKey/pathParam?queryParam : response**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Param | REST | Type | Format | Require |
| dataProtection | queryParam | boolean |  | optional |
| certificateThumbprint | queryParam | string | SHA | optional |

Table 25 Service parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json |  |  | The operation was performed successfully |
| publicCertificate | string |  | PEM |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 403 | application/json | string |  | Not authorized to requested information |
| 404 | application/json | string |  | Public key not found |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 26 Service Response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

## Operation UPLOAD PUBLICKEY

The purpose of the interface is to upload a public key.

### Operation Functionality

No specific functionality for S-421 other than defined by SECOM.

### Operation Parameters

**POST baseUrl/v1/PublicKey { body } : response**

Public certificate x.509 in PEM format, Base64 encoded byte array.

|  |  |  |  |
| --- | --- | --- | --- |
| Body | Type | Format | Require |
| publicCertificate | string | PEM |  |

Table 27 Service body

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HTTP Code | Content | Type | Format | Description |
| 200 | application/json |  |  | The operation was performed successfully |
| 400 | application/json | string |  | Bad request |
| 401 | application/json | string |  | Unauthorized |
| 405 | application/json | string |  | Method not allowed |
| 500 | application/json | string |  | Internal server error |
| 501 | application/json | string |  | Not implemented |

Table 28 Service response

### Values for S-421 Route Exchange Service

No specific values for S-421 other than defined by SECOM.

# Service Dynamic Behaviour

To be discussed: Add information here in this document or refer to IEC 63173-2?

# References

| Nr. |  | Reference |
| --- | --- | --- |
| 1. IALA Guideline G1128 |  | THE SPECIFICATION OF E-NAVIGATION TECHNICAL SERVICES |
| 1. IMO FAL.5 /Circ.52 |  | Guidelines for Harmonized Communication and Electronic Exchange of Operational Data for Port Calls |
| 1. IALA Recommendation R1023 |  | MARITIME RESOURCE NAMES |
| 1. IHO Standard S-100 | 5.2.0 | IHO Universal Hydrographic Data Model  <https://iho.int/uploads/user/pubs/standards/s-100/S-100_5.0.0_Final_Clean_Web.pdf> |
| 1. IEC 63173-1 S-421 Product Specification |  |  |
| 1. NA |  |  |
| 1. IEC 63173-2 SECOM | 1.0.0 |  |
| 1. Service Design – Template SECOM REST |  | G1128 based template for service designs using SECOM REST |
| 1. NIST Digital Signature Standard (DSS–FIPS Publication 186) |  |  |

# Acronyms and Terminology

## Acronyms

|  |  |
| --- | --- |
| Term | Definition |
| API | Application Programming Interface |
| MRN | Maritime Resource Name |
| RTA/RTD | Requested time of arrival/departure |
| URI | Uniform Resource Identifier |
| UUID | Universally Unique Identifier v4 |
| XML | Extendible Mark-up Language |
| XSD | XML Schema Definition |

## Terminology

|  |  |
| --- | --- |
| Term | Definition |
| Operational Node | A logical entity that performs activities. Note: nodes are specified independently of any physical realisation.  Examples of operational nodes in the maritime context are: Maritime Control Center, Maritime Authority, Ship, Port, Weather Information Provider, … |
| Service | The provision of something (a non-physical object), by one, for the use of one or more others, regulated by formal definitions and mutual agreements. Services involve interactions between providers and consumers, which may be performed in a digital form (data exchanges) or through voice communication or written processes and procedures. |
| Service Consumer | A service consumer uses service instances provided by service providers. All users within the maritime domain can be service customers, e.g., ships and their crew, authorities, VTS centres, organizations (e.g., meteorological), commercial service providers, etc. |
| Service Data Model | Formal description of one dedicated service at logical level. The service data model is part of the service specification. Is typically defined in UML and/or XSD. If an external data model exists (e.g., a standard data model), then the service data model shall refer to it: each data item of the service data model shall be mapped to a data item defined in the external data model. |
| Service Interface | The communication mechanism of the service, i.e., interaction mechanism between service provider and service consumer. A service interface is characterised by a message exchange pattern and consists of service operations that are either allocated to the provider or the consumer of the service. |
| Service Operation | Functions or procedure which enables programmatic communication with a service via a service interface. |
| Service Physical Data Model | Describes the realisation of a dedicated service data model in a dedicated technology. This includes a detailed description of the data S-212 to be exchanged using the chosen technology. The actual format of the service physical data model depends on the chosen technology. Examples may be WSDL and XSD files (e.g., for SOAP services) or swagger (Open API) specifications (e.g., for REST services). If an external data model exists (e.g., a standard data model), then the service physical data model shall refer to it: each data item of the service physical data model shall be mapped to a data item defined in the external data model.  In order to prove correct implementation of the service specification, there shall exist a mapping between the service physical data model and the service data model. This means, each data item used in the service physical data model shall be mapped to a corresponding data item of the service data model. (In case of existing mappings to a common external (standard) data model from both the service data model and the service physical data model, such a mapping is implicitly given.) |
| Service Provider | A service provider provides instances of services according to a service specification and service instance description. All users within the maritime domain can be service providers, e.g., authorities, VTS centres, organizations (e.g., meteorological), commercial service providers, etc. |