Service Design for VTS – Vessel Route Exchange Service using SECOM

Contents

[1. Introduction 5](#_Toc190435311)

[1.1. General 5](#_Toc190435312)

[1.2. Intended Readership 5](#_Toc190435313)

[1.3. Inputs from Other Sources 5](#_Toc190435314)

[2. SERVICE DESIGN TEMPLATE IDENTIFICATION 7](#_Toc190435315)

[3. TECHNOLOGY INTRODUCTION 8](#_Toc190435316)

[3.1. General 8](#_Toc190435317)

[3.2. Service technology and service transportation protocol 8](#_Toc190435318)

[3.3. Security 8](#_Toc190435319)

[3.3.1. Communication channel security 8](#_Toc190435320)

[3.3.2. Data protection 8](#_Toc190435321)

[3.3.3. Data Signature 9](#_Toc190435322)

[3.3.4. Data Encryption 9](#_Toc190435323)

[4. SERVICE DESIGN TEMPLATE OVERVIEW 10](#_Toc190435324)

[4.1. General 10](#_Toc190435325)

[4.2. Service Template Interfaces 10](#_Toc190435326)

[4.3. Service Discovery 12](#_Toc190435327)

[5. PHYSICAL DATA MODEL 13](#_Toc190435328)

[5.1. gml:id of the root element 13](#_Toc190435329)

[6. Service Interface Behaviour 13](#_Toc190435330)

[6.1. Common information 13](#_Toc190435331)

[6.1.1. Common parameter values 13](#_Toc190435332)

[6.1.2. Error Codes 14](#_Toc190435333)

[6.2. Applicable SECOM Interfaces 15](#_Toc190435334)

[6.2.1. Upload 15](#_Toc190435335)

[6.2.2. Upload link 16](#_Toc190435336)

[6.2.3. Acknowledgement 16](#_Toc190435337)

[6.2.4. Get 16](#_Toc190435338)

[6.2.5. Get summary 17](#_Toc190435339)

[6.2.6. Get by link 17](#_Toc190435340)

[6.2.7. Access 17](#_Toc190435341)

[6.2.8. Access notification 17](#_Toc190435342)

[6.2.9. Subscription 18](#_Toc190435343)

[6.2.10. Remove Subscription 18](#_Toc190435344)

[6.2.11. Subscription notification 19](#_Toc190435345)

[6.2.12. Capability, Ping, EncryptionKey and PublicKey 19](#_Toc190435346)

[7. SERVICE DYNAMIC BEHAVIOUR 19](#_Toc190435347)

[7.1. Generic sequence for signatures 19](#_Toc190435348)

[7.2. Dynamic Behavior of the Service Interface 21](#_Toc190435349)

[7.2.1. Upload interface 21](#_Toc190435350)

[7.2.2. Upload Link interface 21](#_Toc190435351)

[7.2.3. Get Interface 21](#_Toc190435352)

[7.2.4. Get By Link interface 21](#_Toc190435353)

[7.2.5. Request Access interface 21](#_Toc190435354)

[7.2.6. Subscription interface 21](#_Toc190435355)

[7.2.7. EncryptionKey interface 21](#_Toc190435356)

[7.2.8. Public Key Interface 21](#_Toc190435357)

[8. DEFINITIONS 21](#_Toc190435358)

[8.1. Terminology 21](#_Toc190435359)

[9. ACRONYMS 22](#_Toc190435360)

[10. REFERENCES 22](#_Toc190435361)

[*ANNEX A.* SERVICE DESIGN TEMPLATE OpenAPI (Swagger) in JSON 22](#_Toc190435362)

[*ANNEX B.* Service Instance XML template 22](#_Toc190435363)

Table of figures

**No table of figures entries found.**

List of tables

N/A

# Introduction

This document specifies the design of the route exchange service according to the technical service specification (TODO REF) using SECOM as the transport mechanism for route exchange. This design defines the implementation of the route exchange service that fulfils the requirements defined by IMO TODO REF.

This document was produced as part of the work of IALA VTS committee 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* and is based on the SECOM template found therein.

## General

The purpose of this document is to define a service design using SECOM for the route exchange service defined in the technical service specification. This is done by using SECOM and the S-421 XML representation as defined in the specification. It defines the actual way the interactions and logical interfaces defined in the specification are to be implemented. It also describes how the requirements for authentication etc are to be implemented.

It describes the actual requirements for both ship and shore side systems in order to support route exchange as defined in the specification.

**Note that this design will undergo changes between versions 1.0 and 2.0 as the SECOM standard should be updated during that transition period and the design will be changed to reflect the latest changes. Some of the drafted changes have already been included in this design and as such this design may not be fully compatible with the SECOM 1.0.**

## Intended Readership

This service design description document is intended to be read by service architects, designers and developers of the route exchange service.

## Inputs from Other Sources

This service design follows the requirements, use cases and dynamic behaviour outlined in TODO Ref. The interfaces, parameters and logic of the API is compliant with IEC 63173-2 SECOM and borrows much of its content from there.

“The author thanks the International Electrotechnical Commission (IEC) for permission to reproduce Information from its International Standards. All such extracts are copyright of IEC, Geneva, Switzerland. All rights reserved. Further information on the IEC is available from www.iec.ch. IEC has no responsibility for the placement and context in which the extracts and contents are reproduced by the author, nor is IEC in any way responsible for the other content or accuracy therein.”[[1]](#footnote-2)

# SERVICE DESIGN TEMPLATE IDENTIFICATION

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

1. Service Design Identification

|  |  |
| --- | --- |
| Name | VTS – Vessel Route Exchange using SECOM Technical Service Design |
| Implements | VTS – Vessel Route Exchange Technical Service Specification 0.7  urn:mrn:iala:techsvc:ss:vts-res:0.7 |
| ID | urn:mrn:iala:techsvc:design:vts-res-secom:0.1 |
| Version | 0.1 |
| Description | The VTS – Vessel Route Exchange Service using SECOM design describes how to implement a Route Exchange Service using SECOM as the transport mechanism between ship and VTS in the exchange of route plans. |
| Keywords | REST, SECOM, VTS, MS1, Route Exchange, Ship Traffic Management, S-421 |
| Architect(s) | Ramin Miraftabi |
| Status | provisional |

# TECHNOLOGY INTRODUCTION

## General

This design template realizes the service specification (TODO reference) using SECOM as defined in IEC 63173-2 (TODO reference).

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 service 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 SECOM REST 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 1.2 and 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 4.3.4 SECOM data protection**

**Reference: IHO 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 with the receiver.

While sending standalone S-421 documents without the additional metadata of datasets or exchange sets is preferred, if datasets or exchange sets are used the data (one or more data files) can optionally be packaged and compressed before being signed and sent.

### Data Signature

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

**Reference: IHO S-100 ed5.0.0 Part 15-8 Data Authentication**

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

The algorithm for signing data must be ECDSA-384-SHA2.

The signature is transported in HEX.

### Data Encryption

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

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

The encryption algorithm for encryption is AES and CBC mode.

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

# SERVICE DESIGN TEMPLATE OVERVIEW

## General

The design uses SECOM defined APIs. As such, it is important to understand that both the route exchange service and its consumers must be able to function as client and server as understood in the traditional HTTP world. As such from here on when the term service is used, it applies to the route exchange service and the consumer is the ship system that is communicating VTS.

This design does not concern itself with how the service will communicate with the VTS system. The instances of this design may be developed as components directly integrated with the VTS system or as independent microservices that communicate with the VTS system via different integration mechanisms e.g., APIs or by emitting and consuming events.

## Service Template Interfaces

**Reference: IEC 63173-2 SECOM v1.0.0 Clause 5.7** **SECOM service interface definitions**

SECOM does not require that all the interfaces defined in the standard must be implemented. Thus, for the purposes of this service and its consumers, only the following interfaces are required:

|  |  |  |
| --- | --- | --- |
| Interface | SECOM Reference | Comment |
| Upload | IEC 63173-2 SECOM v1.0.0 Clause 5.7.2 service interface – Upload | This interface is called when the client uploads (pushes) a route to the server. The sender decides the protection of the route. This interface is provided by both the consumer and service. |
| Upload Link | IEC 63173-2 SECOM v1.0.0 Clause 5.7.3 service interface – Upload Link | This interface is called when the client uploads (pushes) a reference pointer to a route that is larger than what the server allows via HTTP POST. The data is downloaded using interface Get By Link. This interface is provided by both the consumer and service. |
| Acknowledgement | IEC 63173-2 SECOM v1.0.0 Clause 5.7.4 service interface – Acknowledgement | This interface is called as response to Acknowledgement request in the previous request. This interface is provided by both the consumer and service. |
| Get | IEC 63173-2 SECOM v1.0.0 Clause 5.7.5 service interface – Get | This interface is called when the client gets (pulls) data from the service. This interface must be provided by the consumer, but supporting get-requests from the consumer is optional for the service. |
| Get Summary | IEC 63173-2 SECOM v1.0.0 Clause 5.7.6 service interface – Get Summary | This interface is called when the client gets a summary of available data from the service. The data is retrieved (pulled) using the interface Get. |
| Get By Link | IEC 63173-2 SECOM v1.0.0 Clause 5.7.7 service interface – Get By Link | This interface is called when the client downloads (pulls) a large route by reference given from interface Upload Link. This interface is provided by both the consumer and service. |
| Access | IEC 63173-2 SECOM v1.0.0 Clause 5.7.8 service interface – Access | This interface is called when the client asks for access to data from the service. Response is given by callback to Access Notification. This interface is provided by the. |
| Access Notification | IEC 63173-2 SECOM v1.0.0 Clause 5.7.9 service interface – Access Notification | This interface is called as response to interface Access. This interface is provided by the service. |
| Subscription | IEC 63173-2 SECOM v1.0.0 Clause 5.7.10 service interface – Subscription | This interface is called when the client or server initiates subscription on data from the service. Response is given with interface Upload and Subscription Notification. This interface is provided by both the consumer and service. |
| Remove Subscription | IEC 63173-2 SECOM v1.0.0 Clause 5.7.11 service interface – Remove Subscription | This interface is called when the client or server removes subscription. Response is given with interface Subscription Notification. This interface is provided by both the consumer and service. |
| Subscription Notification | IEC 63173-2 SECOM v1.0.0 Clause 5.7.12 service interface – Subscription Notification | This interface is called as response from Subscription or Remove Subscription. This interface is provided by both the consumer and service. |
| Capability | IEC 63173-2 SECOM v1.0.0 Clause 5.7.13 service interface – Capability | This interface is called when the client asks for the service capabilities. This interface is provided by both the consumer and service. |
| Ping | IEC 63173-2 SECOM v1.0.0 Clause 5.7.14 service interface – Ping | This interface is called when the client checks the availability of the service. This interface is provided by both the consumer and service. |
| EncryptionKey | IEC 63173-2 SECOM v1.0.0 Clause 5.7.15 service interface – EncryptionKey | This interface is called when sending (pushing) encryption key to a receiver. This interface is provided by both the consumer and service. |
| PublicKey | IEC 63173-2 SECOM v1.0.0 Clause 5.7.16 service interface – PublicKey | This interface is called when the client gets (pulls) the public certificate(s) from the service. This interface is provided by both the consumer and service. |

All unused interfaces of SECOM should be implemented to return HTTP 501 as specified in TODO SECOM REF.

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 VTS system or the vessel and the vessel’s shoreside SECOM interface. These are specific for each implementation and depend on the VTS system and vessel’s system.

## Service Discovery

Services implemented according to this design must submit their instance description to a valid service registry that follows the Maritime Service Registry definition TODO Ref.

An XML template for the instance description is provided as an annex to this design TODO Ref.

# 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 TODO Ref section 5.

The S-421 data that is used by the service must be a valid XML document according to the S-421 schema TODO ref and follow the requirements defined in the service specification TODO Ref. The service specification has extended requirements on what constitutes a valid route in the context of this service and the different use cases it is designed to support.

While the S-421 schema supports S-100 datasets and exchange sets, the more minimal version of the document without dataset information is preferred.

Implementing services should send the route without dataset or exchange set information.

Services must support sending datasets or exchange sets if requested by client.

## gml:id of the root element

The root element of the XML document is required to have a gml:id attribute. To allow better tracing of the shared document as a transaction it must be connectable to the transactionIdentifier of the calls. However, as the transactionIdentifier is required by SECOM to be a UUID which may begin with a number and gml:id is an xml:id which must begin with a letter, the following requirement is placed:

The gml:id of the dataset element of the route must follow the format: urn:uuid:[uuid] where uuid is the UUID also used in the transactionIdentifier of the SECOM calls.

# Service Interface Behaviour

As defined by SECOM, all communication between the components of the service that are in the scope of this design document is done via REST calls.

Additional information on the interfaces and parameters is available in the attached OpenAPI template of service in Annex TODO Ref

## Common information

For all parameters that are not set they should be omitted or can passed with a null value.

### Common parameter values

For all interfaces, the following parameter values must be used when applicable:

* containerType – should always be “NONE” (i.e. 2)
* dataProductType – must always be “S421”
* exchangeMetadata
  + protectionScheme – must always be “SECOM”
* callbackEndpoint – The endpoint of the consumer’s SECOM service that accepts the interface calls. The endpoint must be defined in a way that appending the SECOM standard REST endpoints (e.g. /v2/subscription) produces a valid URL. If the callbackEndpoint is not a part of the subscription, the service must search for the vessel’s SECOM service from a service registry

### Error Codes

If incoming route does not pass schema validation, the SECOM\_ResponseCode must be 3.

If incoming route does not pass business rule validation, the SECOM\_ResponseCode must be 0.

If certificate provided is invalid, the SECOM\_ResponseCode must be 2.

If the signature verification fails, the SECOM\_ResponseCode must be 1.

In all of these cases, the response returned to the client must have the HTTP status 400.

## Applicable SECOM Interfaces

The Capability and Ping interfaces are not discussed in this design as they follow the requirements defined in SECOM TODO Ref and need no further elaboration here.

All interface descriptions below only have the information necessary for the business logic of this service. All definitions and descriptions that are directly derived from the SECOM standard left to be read from the SECOM standard and the API documentation template in TODO create Annex.

All SECOM communication is done via REST APIs with JSON as the data format for the SECOM data and XML for the S-421 data.

For all operations only those parameters that have predefined values or their usage needs to be explained are described below. Otherwise refer to SECOM TODO ref.

### Upload

The Upload interface must be available both on the service as well as the SECOM interface of the vessel.

For all parameters, follow the requirements specified in the general section and:

* fromSubscription –
  + must always be true when sent from service to vessel as service never uploads without subscription;
  + must always be false when route is initially sent from ship system to shore;
  + must always be true when ship is sending updates to route following the subscription for updates from shore.
* ackRequest – By default, set to 3 (both delivered and opened) unless:
  + ship system has not requested acknowledgements of route opening when sending updates as part of subscription from shore. It is set to 1 (delivered) in this case.
  + Route is moved to monitoring. Set to 1 in this case.
* data – S-421 route that follows the requirements specified in TODO Ref with the error codes specified above.

The return value of the upload interface must follow the SECOM standard. The message is not expected and should only be returned in the case of an error to provide additional information is available.

If the uploaded route is too large the returned HTTP status code must be 413. The route must then be shared using the upload link interface.

The vessel may return HTTP status 403 to any route it receives that it has not requested or has not previously shared with shore.

Shore systems should not return HTTP status 403 (Not authorized) to vessels.

### Upload link

The Upload link interface must be available both on the service as well as the SECOM interface of the vessel.

This is used when the route shared is too large to share via the upload interface and a HTTP status 413 has been received. If the sender knows the route will be too large due to previous interactions, then it may be used directly.

It must not be used as the first method of interaction between consumer and service and any upload link that is received that is not a part of an established route exchange initiated by at least one call to the upload interface must be rejected and HTTP status code 403 must be returned.

For all parameters, follow the requirements specified in the general section and what is specified for upload.

See the Get by link interface definition below for how the retrieval of the data is handled.

### Acknowledgement

There are no specific business rules that must be defined here for the acknowledgment interface that are not covered by the SECOM API specification document.

The requested level of acknowledgment is specified in other parts of this document.

### Get

The Get interface must be available for the consumer and may be available in the service if the reference routes use case has been implemented.

The consumer uses the get interface to retrieve reference routes from a route library.

The service uses the get interface to retrieve the current route from the ship once the route has been initially shared by the ship (use case 15).

When used by the consumer use the parameters specified in the general section and:

* dataReference – used when the reference of the reference route is known, and the latest version of the route is being fetched
* geometry – used when searching for reference routes that intersect a geometry
* unlocode – used to search for reference routes that intersect with the unlocode, generally either starting or ending at location
* validFrom and validTo – used to specify the time range in which the reference route must be valid. The times must include a time zone and should be in UTC.

Return values must follow what is specified in SECOM.

When used by the service use the parameters specified in the general section and:

* dataReference – must be the uuid part of the gml:id of the route.

When the service is retrieving an update to the route, no other parameters are allowed and if passed the consumer should respond with HTTP status code 400.

If the route is requested by service without it being previously shared (use case 10 in specification) then the consumer must respond with HTTP status code 403.

### Get summary

The Get summary interface may be available on the service if the reference route library functionality has been implemented.

The interface is used instead of Get to get a list of available routes if the consumer so wishes.

All parameters and their usage should follow what was previously described for the get interface.

### Get by link

The Get by link interface must be available both on the service as well as the SECOM interface of the vessel.

This is used when the route shared is too large to share via the upload interface and the route has been shared via the upload link interface.

### Access

The Access interface must be supported by the consumer.

The interface is used when VTS is requesting the route from the vessel without the route being shared previously according to use case 10 in the specification. Calling this interface requires that the vessel is discoverable in a maritime service registry.

The vessel is not required to share its route and may respond with HTTP status code 403 if not allowed.

For all parameters, follow the requirements specified in the general section and:

* dataReference must not be set
* reason – must be provided and contain rationale for the request
* reasonEnum – must be set to 0 (required by authority)

If the vessel’s SECOM interface is able to determine automatically that access is not granted (e.g. based on the identity of the requester) then HTTP status code 403 may be returned and the mariner aboard the vessel need not be notified of the request.

### Access notification

The Access interface must be supported by the service.

The interface is used when the mariner onboard the vessel has been notified of the request by VTS to receive the route.

The following use of the parameters is required:

* transactionIdentifier – if access is granted follows the rules described in the general section, otherwise is a unique UUID.
* decision and decisionReason – according to SECOM specification.

### Subscription

The subscription interface must be implemented in both the service and consumer.

The following parameters of the SubscriptionRequestObject are expected to be passed as a part of the request:

* dataReference – UUID that is the suffix of the gml:id dataset element in the route that was shared.
* productVersion – must be left unset. The service specification defines the allowed version of S-421 that the service and consumer must support.
* geometry – must not be provided by the consumer, may be provided by the service.
* unlocode – must not be provided
* subscriptionPeriodStart – may be provided, if not provided or in the past, the recipient will default to time of reception of call to interface
* subscriptionPeriodEnd – must not be provided. Both the consumer and the service must end subscriptions.

The returned SubscriptionResponseObject contains the following:

* message – optional. Consumers may or may not support this.
* subscriptionIdentifier – must have an UUID that can be used to remove the subscription once communication is no longer necessary.

The service must use the MRN of the vessel retrieved from the certificate submitted in the SECOM request headers to identify the vessel for which the subscription is created. The vessel must not receive routes intended for other vessels because of this subscription.

The vessel must also ensure that it does not send route updates to those shore systems that have not subscribed to updates or send route updates intended for another shore system to the incorrect shore system.

### Remove Subscription

The remove subscription interface must be implemented in both the service and consumer.

The removeSubscriptionObject must have the subscriptionIdentifier returned when the subscription was created.

The consumer should call removeSubscription when

1. The route changes and no longer passes through a VTS area and the subscription for that VTS area can be removed.
2. The voyage has been completed and no further sharing of the route is necessary.
3. The route is cancelled.

The above causes assume that the ship system creates a new subscription for each route that is shared. If the ship systems create an open-ended subscription for route sharing (e.g. vessel is on a regular route and subscriptions are created for each VTS area) then the subscription can be kept open as long as necessary.

The service must call removeSubscription when the consumer ends its subscription as the sharing of the route is no longer needed and no more updates to the route from the consumer are to be expected.

### Subscription notification

The subscription interface must be implemented in both the service and consumer.

The service is not required to call the subscriptionNotification of the consumer unless the creation or removal of the subscription requires asynchronous action or is the result of a maintenance operation.

The consumer should call the subscriptionNotification of the service in most cases as the recipient of the request from the service is typically not the ship system that is required to know about the subscription for the sending of route updates. If the subscription and removeSubscription are synchronous processes when called from VTS to ship system, subscriptionNotification is not required.

Use of the interface is according to the SECOM specification.

### Capability, Ping, EncryptionKey and PublicKey

The interfaces Capability, Ping, EncryptionKey and PublicKey are not discussed here as they follow the requirements defined in IEC 63173-2 SECOM TODO Reference) and need no further elaboration here.

# SERVICE DYNAMIC BEHAVIOUR

In the following diagrams and descriptions, the communication between the vessel’s onboard systems and its SECOM service is out of scope for the purposes of this design.

Similarly, the communication and interfaces between the service and the VTS system is out of scope of this design

## Generic sequence for signatures

This sequence assumes the following actors:

Data Provider 🡪 Data Sender 🡪 Data Receiver 🡪 Data Consumer

The sequence describes the data from its source through transport using SECOM to the end-user.

Provider preparation:

* The data provider must have a valid maritime identity according to G1183 (TODO ref)
* The data provider creates a private-public key pair
* The data provider sends the public key to the MIR to get it signed and connected to a specific identity/identifier in the MIR (Certificate Signing Request)
* The data sender must have a valid maritime identity according to G1183
* The data sender creates a private-public key pair
* The data sender sends the public key to the MIR to get it signed and connected to a specific identity/identifier in the MIR (Certificate Signing Request)

Data Provider and Data Sender:

* The data provider selects identifier and certificate to use for the data signing
* The data provider creates a signature for the data
* The data provider transfers the data with its signature to the data sender.
* The data sender protects the data transferring by the selected communication mechanism, e.g. SECOM. SECOM Upload wraps the data, the signature and metadata into an envelope. The envelope is signed by the data sender.
* The data sends contacts the data receiver and transfers the data

Data Receiver and Data Consumer

* The data receiver receives the transferred object with envelope. The envelope signature is verified.
* The data receiver forwards the data and data signature to the data consumer
* The data consumer verifies the data signature

## Dynamic Behavior of the Service Interface

The following chapters contains sequence diagrams for the service interfaces. This section expands upon the dynamic behavior described in the service specification.

### Upload interface

### Upload Link interface

### Get Interface

### Get By Link interface

### Request Access interface

### Subscription interface

### EncryptionKey interface

### Public Key Interface

# DEFINITIONS

The definitions of terms used in this IALA Guideline can be found in the International Dictionary of Marine Aids to Navigation (IALA Dictionary) at <http://www.iala-aism.org/wiki/dictionary> and were checked as correct at the time of going to print. Where conflict arises, the IALA Dictionary shall be considered as the authoritative source of definitions used in IALA documents.

## Terminology

Persons producing the Technical Service are invited to add definitions to the following list as appropriate.

1. Definition of terminology – Technical Service

| Term | Definition |
| --- | --- |
| **External Data Model** | Describes the semantics of the ‘maritime world’ (or a significant part thereof) by defining data structures and their relations. This could be at logical level (e.g. in UML) or at physical level (e.g. in XSD schema definitions), as for example standard data models, or S-100 based data produce specifications. |
| **Message Exchange Pattern** | Describes the principles two different parts of a message passing system (in our case: the service provider and the service consumer) use to interact and communicate with each other. Examples:  In the Request/Response MEP, the service consumer sends a request to the service provider to obtain certain information; the service provider provides the requested information in a dedicated response.  In the Publish/Subscribe MEP, the service consumer establishes a subscription with the service provider to obtain certain information; the service provider publishes information (either in regular intervals or upon change) to all subscribed service consumers. |

# ACRONYMS

Persons producing the Technical Service are invited to provide a list of acronyms as appropriate.

|  |  |
| --- | --- |
| **REST** | REpresentational State Transfer |
| **SECOM** | Secure Communication |
| **MIR** | Maritime Identity Registry |
| **MSR** | Maritime Service Registry |
|  |  |

# REFERENCES

1. IALA Guidelineline 1128 11?? on Specification of e-Navigation Technical Services
2. IEC 63173-2 SECOM v1.0.0
3. SECOM Service Interface OpenAPI (https://cirm.org/secom)
4. IHO S-100 ed 5.2.0
5. NIST Digital Signature Standard (DSS–FIPS Publication 186)
6. RFC 9110 HTTP Semantics
7. SERVICE DESIGN TEMPLATE OpenAPI (Swagger) in JSON

TODO OpenAPI template ref

1. Service Instance XML template

TODO XML template ref

1. IEC 63173-2 ed. 1.0 “Copyright © 2022 IEC Geneva, Switzerland. www.iec.ch” [↑](#footnote-ref-2)