Chapter IV
Cookbook
Version 1.5

This document is provided to you free, of charge, by the

eHealth platform

Willebroekkaai 38 – 1000 Brussel
38, Quai de Willebroek – 1000 Bruxelles

Table of contents

Contents

To the attention of: “IT expert” willing to integrate this web service.

1. Document management

1.1 Document history

Version	Date	Author	Description of changes / remarks
1	24/05/2011	eHealth platform	First version
1.1	01/12/2011	eHealth platform	Update
1.2	16/01/2015	eHealth platform	Update hyperlinks in French and Dutch
1.3	19/12/2019	eHealth platform	ETk – Addition information
1.4	03/08/2022	eHealth platform	§ 2.3 eHealth document references (updated) § 3 Support (added) § 4.1.1 WS-I Basic profile (added) § 4.1.2 Tracing (added)
1.5	12/05/2025	eHealth platform	Add new service dedicated to pharmacies

2. Introduction

2.1 Goal of the service

The eHealth platform makes available to the medical advisors of the healthcare insurance organization and the caregivers dedicated services to the medical agreements 'Chapter IV' precisely:

- The demand for the medical advisor agreements.
- The consultation of the medical advisor agreements.

2.2 Goal of the document

This document is not a development or a programming guide for internal applications. Instead, it provides functional and technical information and allows an organization to integrate and use the eHealth service.

However, in order to interact in a smooth, homogeneous and risk controlled way with a maximum of partners, eHealth partners must commit to comply with the requirements of specifications, data format and release processes described in this document.

Technical and business requirements must be met in order to allow the integration and validation of the eHealth service in the client application.

Detailed description of the functionality of the service, semantics of the particular elements and other general information about the service are out of the scope of this document. This kind of information can be found in the documentation provided by MyCareNet on their Sharepoint.

2.3 eHealth platform document references

On the portal of the eHealth platform, you can find all the referenced documents.[1]. These versions, or any following ones, can be used for the eHealth platform service.

ID	Title	Version	Date	Author
1	SOA – Error guide	1.0	10/06/2021	eHealth platform
2	Cookbook Secure Token Service – WS Trust	1.1	12/01/2024	eHealth platform
3	MyCareNet ChapterIV SSO	1.5	12/05/2025	eHealth platform
4	End-to-End Encryption Known recipient	2.9	18/07/2022	eHealth platform
5	End-To-End Encryption Unknown recipient	1.7	19/07/2022	eHealth platform

[1] ***www.ehealth.fgov.be/ehealthplatform***

2.4 External document references

All the MyCareNet documentation can be found on their Sharepoint[2]. The documentation referenced in this section may change over time.

All documents can be found through the internet. They are available to the public, but not supported by the eHealth platform.

ID	Title	Source	Date	Author
1	Basic Profile Version 1.1	http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html	24/08/2004	Web Services Interoperability Organization
2	MCN – Chap IV functional description	N/A	N/A	CIN

[2] In order to have access to the Sharepoint, you need to create an account which can be requested at : ***https://fra.mycarenet.be/mycarenet/utilisateurs-de-mycarenet*** or ***https://ned.mycarenet.be/algemene-beschrijving/gebruikers***

3. Support

3.1 Helpdesk eHealth platform

3.1.1 Certificates

In order to access the secured eHealth platform environment you have to obtain an eHealth platform certificate, used to identify the initiator of the request. In case you do not have one, please consult the chapter about the eHealth Certificates on the portal of the eHealth platform

- ***https://www.ehealth.fgov.be/ehealthplatform/nl/ehealth-certificaten***
- ***https://www.ehealth.fgov.be/ehealthplatform/fr/certificats-ehealth***

For technical issues regarding eHealth platform certificates

- Acceptance: ***acceptance-certificates@ehealth.fgov.be***
- Production: ***support@ehealth.fgov.be***

3.1.2 For issues in production

eHealth platform contact centre:

- Phone: 02 788 51 55 (on working days from 8 am till 6 pm)
- Mail: ***support@ehealth.fgov.be***
- *Contact Form :*
  - ***https://www.ehealth.fgov.be/ehealthplatform/nl/contact*** (Dutch)
  - ***https://www.ehealth.fgov.be/ehealthplatform/fr/contact*** (French)

3.1.3 For issues in acceptance

Integration-support@ehealth.fgov.be

3.1.4 For business issues

- regarding an existing project: the project manager in charge of the application or service
- regarding a new project or other business issues: ***info@ehealth.fgov.be***

3.2 Status

The website ***https://status.ehealth.fgov.be*** is the monitoring and information tool for the ICT functioning of the eHealth services that are partners of the Belgian eHealth system.

3.3 Support desk – contact points CIN/NIC

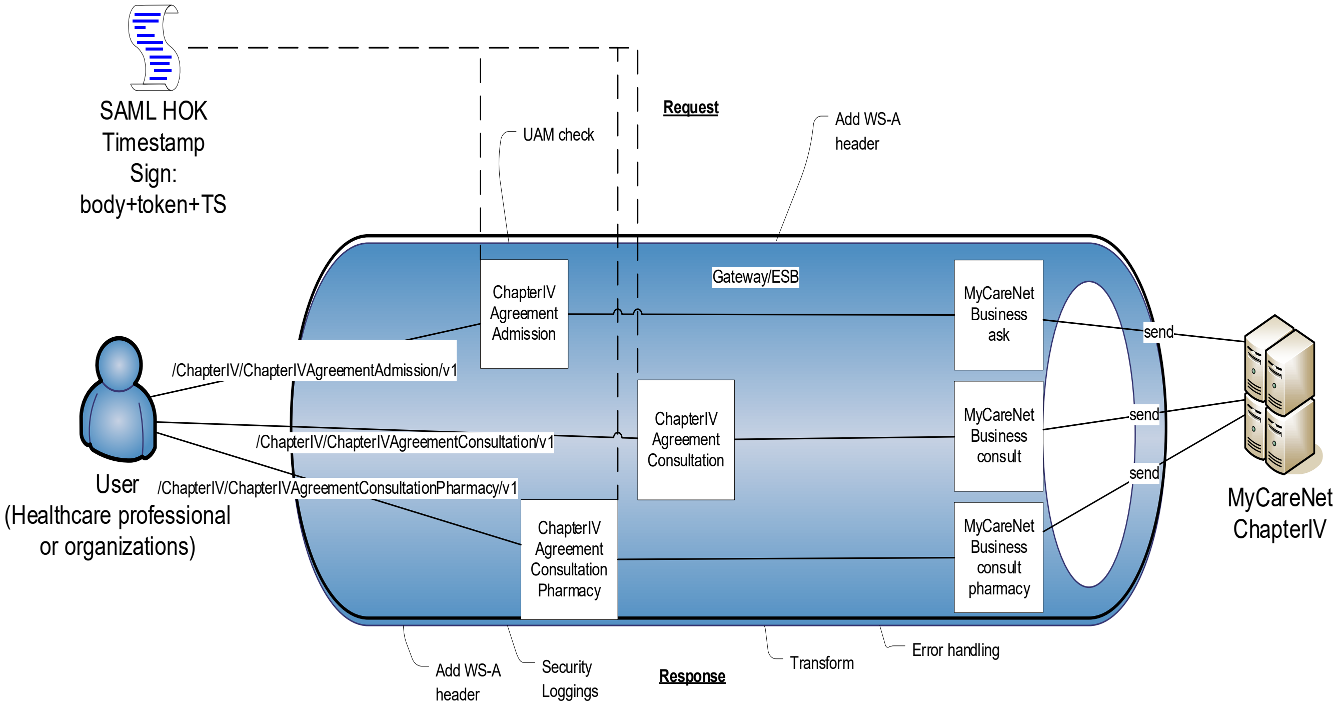
3.3.1 MyCareNet Helpdesk:

- Telephone: 02 891 72 56
- Mail: ***support@intermut.be***

3.3.2 Technical contact center MyCareNet:

- Telephone: 02 431 47 71
- Mail: ***ServiceDesk@MyCareNet.be***

4. Global overview


SAML HOK
Timestamp
Sign:
body+token+TS
UAM check
Request
Add WS-A header
Gateway/ESB
ChapterIV Agreement Admission
MyCareNet Business ask
/ChapterIV/ChapterIVAgreementAdmission/v1
/ChapterIV/ChapterIVAgreementConsultation/v1
ChapterIV Agreement Consultation
MyCareNet Business consult
User
(Healthcare professional or organizations)
/ChapterIV/ChapterIVAgreementConsultationPharmacy/v1
ChapterIV Agreement Consultation Pharmacy
MyCareNet Business consult pharmacy
send
send
send
MyCareNet ChapterIV
Add WS-A header
Security Loggings
Response
Transform
Error handling

The MyCareNet ChapterIV services are secured with the SAML HOK policy. Therefore, prior to call the services, a SAML token must be obtained via the eHealth STS. The obtained token must be then included in the header of the request message, where the timestamp and the body must be signed with the certificate as used in the HOK profile of the SAML token (more detailed technical description can be found in section 5 Step-by-step of this cookbook). The body contains the ChapterIV request.

The eHealth Gateway (Gateway/ESB) verifies the security (authentication, authorization, etc.) and forwards the request to MyCareNet. Then, the service returns the response delivered by the MyCareNet backend.

Note:

In some cases, the eHealth ESB executes a call out to AttributeAuthority service to verify the therapeutic link between the healthcare provider and the patient. This is described in more details in section 5.2.

5. Step-by-step

5.1 Technical requirements

In order to be able to test the MyCareNet ChapterIV services, you need to take the following steps:

1. **Create a test case:** If the testing is done for a real care provider, the real NIHII number of the care provider can be used. Otherwise, you will receive a test NIHII number from the eHealth development team (you must indicate the service called and the kind of profile needed). You always need to request the configuration of the test cases to eHealth (*info@ehealth.fgov.be*).
2. **Request an eHealth test certificate:** a test certificate must be requested to eHealth.

   (https://www.ehealth.fgov.be/ehealthplatform/fr/data/file/view/fe321dd6989a5b71f05f0d11fd92fe5db21bcfcc?name=Procuration%20Form%20eHealthTestCert.pdf).
3. **Obtain the SAML token from the STS:** the eHealth test certificate obtained in the previous step is used for identification at the STS and as the Holder-Of-Key (HOK) certificate.
4. **Call the ChapterIV web services**.

The rules to access the ChapterIV are the same in acceptation as in production.

Access rules:

- authentication with a care providers certificate (see § 3.1 for the information on the certificates, and further in this section for the information about the SAML token).
- authentication with the certificate of a mandate holder (see § 3.1 for the information on the certificates, and further in this section for the information about the SAML token).

In order to implement a WS call protected with a SAML token, you can reuse the implementation as provided in the "eHealth technical connector". Nevertheless, eHealth implementations use standards and any other compatible technology (WSstack for the client implementation) can be used instead.

- ***https://www.ehealth.fgov.be/ehealthplatform/nl/service-ehealth-platform-services-connectors***
- ***https://www.ehealth.fgov.be/ehealthplatform/fr/service-ehealth-platform-services-connectors***

Alternatively, you can write your own implementation. The usage of the STS and the structure of the exchanged xml-messages are described in the eHealth STS – WS Trust cookbook.

5.1.1 Use of the eHealth SSO solution

This section specifies how to call the STS in order to have access to the WS. You must precise several attributes in the request. The details on the identification attributes and the certification attributes can be found in the separate document MyCareNet ChapterIV SSO.

To access the MyCareNet ChapterIV WS, the response token must contain “true” for all of the ‘boolean’ certification attributes and a non-empty value for other certification attributes.

If you obtain “false” or empty values, contact the eHealth platform to verify that they correctly configured the requested test case.

5.1.2 Encryption

All the information about the use of the encryption libraries and the call to the ETK (eHealth Token Key) depot are described in the End-To-End Encryption (ETEE) cookbooks on the eHealth portal.

To encrypt the request parts, you have to call the GetEtk operation to pick up the right ETK from the eHealth ETK depot. By example, the table below provides you the identifiers to use in the GetEtkRequest.

Environment	Type	Value	Application ID
Integration Test Environment	CBE	0820563481	MYCARENET
Acceptance Environment	CBE	0820563481	MYCARENET
Production Environment	CBE	0820563481	MYCARENET

The encryption to a HIO (unknown recipient encryption) is done with a symmetric key as obtained from the KGSS. In order to allow any HIO (but only a HIO) to decrypt the message, the key has to be requested with the allowed-reader specified with the following arguments:

- **Namespace:** urn:be:fgov:certified-namespace:ehealth
- **Name:** urn:be:fgov:kbo-bce:organization:cbe-number:ehealth:1.0:hio:boolean
- **Value:** true

For example:

```
<GetNewKeyRequestContent xmlns="urn:be:fgov:ehealth:etee:kgss:1_0:protocol">
        <AllowedReader>
                <Namespace>urn:be:fgov:certified-namespace:ehealth</Namespace>
                <Name>urn:be:fgov:kbo-bce:organization:cbe-number:ehealth:1.0:hio:boolean</Name>
                <Value>true</Value>
        </AllowedReader>
        <ETK>MIAGCS...</ETK>
</GetNewKeyRequestContent>
```

5.1.3 Security policies to apply

We expect that you use SSL one way for the transport layer.

To call the ChapterIV WS:

- Add the business message to the soap body
- Add to the SOAP header the following elements:
  - **SAML Token**: The SAML assertion received from the eHealth STS. This assertion needs to be forwarded exactly as received in order to not to break the signature of the eHealth STS. The token needs to be added accordingly to the specifications of the OASIS SAML Token Profile (HOK)).
  - **Timestamp**.

  - A **signature** that has been placed on the SOAPBody and the timestamp with the certificate of which the public key is mentioned in the SAML Assertion.
- The signature element (mentioned above) needs to contain:
  - SignedInfo with References to the SOAPBody and the Timestamp.
  - KeyInfo with a SecurityTokenReference pointing to the SAML Assertion.

See also the WSSP in the WSDL[3] (also included in the documentation).

5.1.4 WS-I Basic Profile 1.1

Your request must be WS-I compliant (See Chap 2.4 - External Document Ref).

5.1.5 Tracing

To use this service, the request SHOULD contain the following two http header values (see RFC ***https://datatracker.ietf.org/doc/html/rfc7231#section-5.5.3***):

1. User-Agent: information identifying the software product and underlying technical stack/platform. It MUST include the minimal identification information of the software such that the emergency contact (see below) can uniquely identify the component.
   a. Pattern: {minimal software information}/{version} {minimal connector information}/{connector-package-version}
   b. Regular expression for each subset (separated by a space) of the pattern: [[a-zA-Z0-9-\/]*\/[0-9azA-Z-_.]*
   c. Examples:

      User-Agent: myProduct/62.310.4 Technical/3.19.0

      User-Agent: Topaz-XXXX/123.23.X freeconnector/XXXXX.XXX
2. From: email-address that can be used for emergency contact in case of an operational problem.

   Examples:

   From: ***info@mycompany.be***

[3] *WSDL's can be found in the eHealth Service Registry: https://www.ehealth.fgov.be/ehealthplatform/nl/service-api-catalog*

5.2 Therapeutic link verification

As explained previously, a healthcare actor can consult the agreement of a patient if and only if it exists a therapeutic link between them. The verification is only applicable:

- For physician for the service ChapterIVAgreementConsultation.
- For pharmacy for the service ChapterIVAgreementConsultationPharmacy

This verification is made in the ESB eHealth.

If there is no existing therapeutic link, the WSC receives an error SOA-01002 (see section 8 for more details about the SOA errors)

5.3 Web service

As for now, only the operations described below are available (when support for new user types is added, additional operations will be added to the service). The operations are grouped in the following services:

- Chap4AgreementConsultation Webservice
  - consultChap4MedicalAdvisorAgreement
- Chap4AgreementConsultationPharmacy Webservice
  - consultChap4MedicalAdvisorAgreement
- Chap4AgreementAdmission Webservice
  - askChap4MedicalAdvisorAgreement

The remainder of this section describes the structure of the business request messages.

The operation consultChap4MedicalAdvisorAgreement is described in section 5.3.1.

Section 5.3.2 describes the common element types used in these structures and in the structures of the response types.

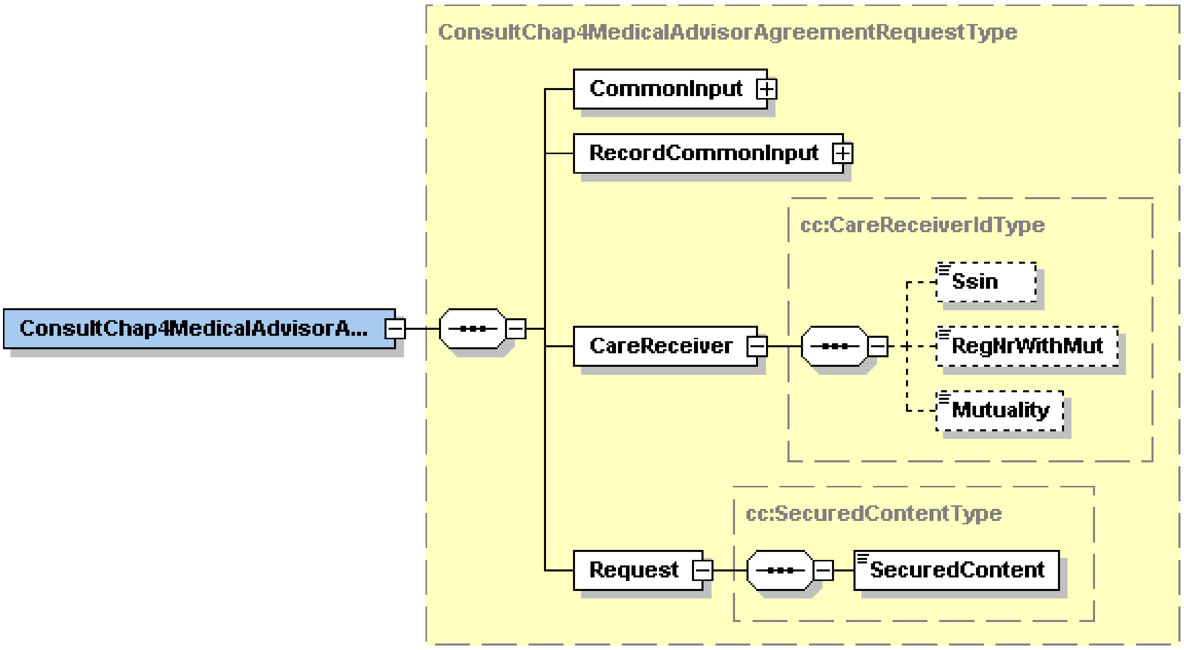
For more detail on the specific elements and the concepts behind them, see the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.1 ConsultChap4MedicalAdvisorAgreement

5.3.1.1 Input arguments in ConsultChap4MedicalAdvisorAgreementRequest

This section only describes the structure of the message. For the business description, see the documentation as provided by CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive).

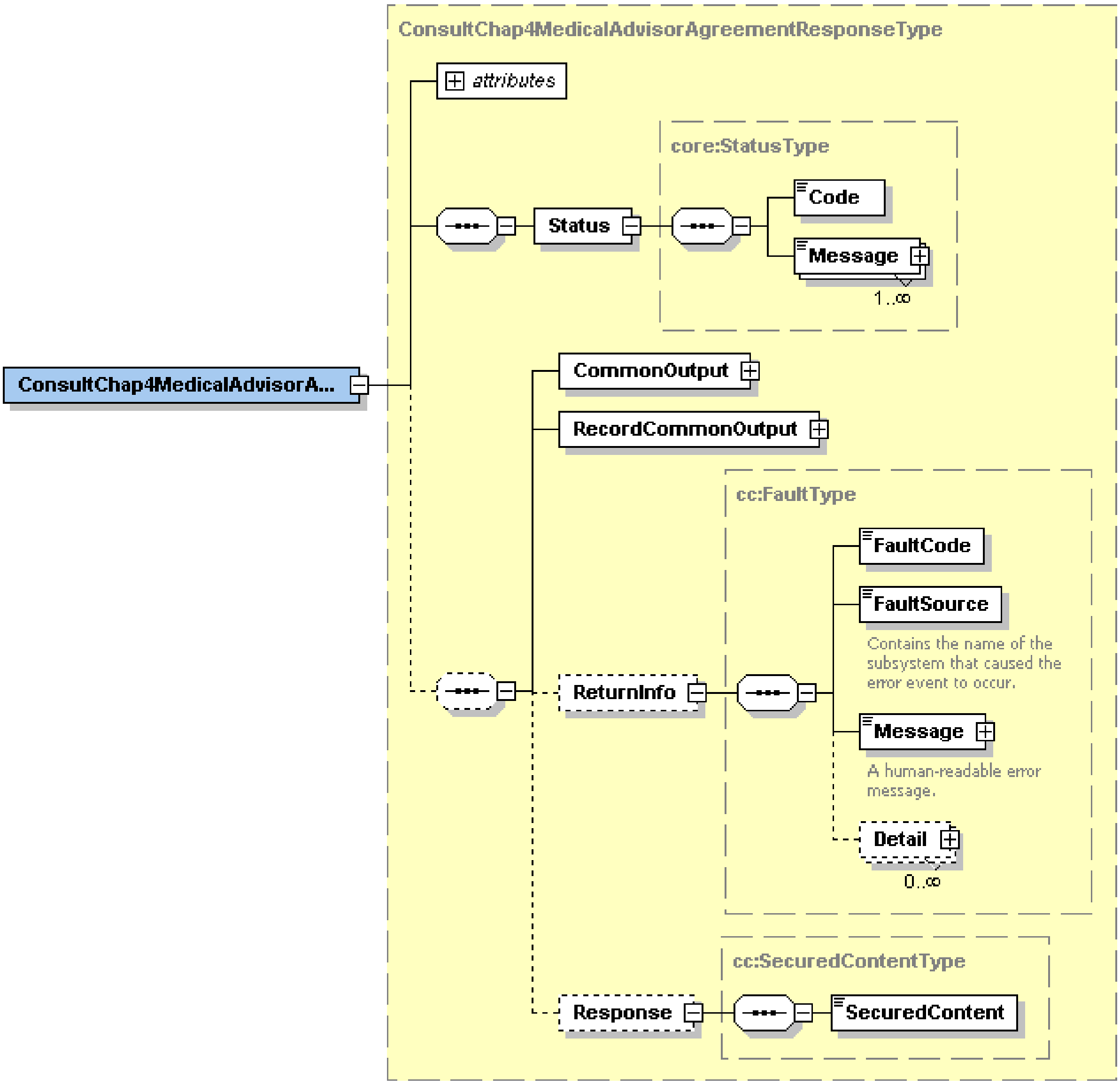
The ConsultChap4MedicalAdvisorAgreement request has the structure as shown on the below figure:



Field name	Descriptions
CommonInput	See section 5.3.3.1 : CommonInputType
RecordCommonInput	See section 5.3.3.2: RecordCommonInputType
CareReceiver	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)
Request	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.1.2 ConsultChap4MedicalAdvisorAgreement response

The ConsultChap4MedicalAdvisorAgreement response has the structure as shown on the below figure:

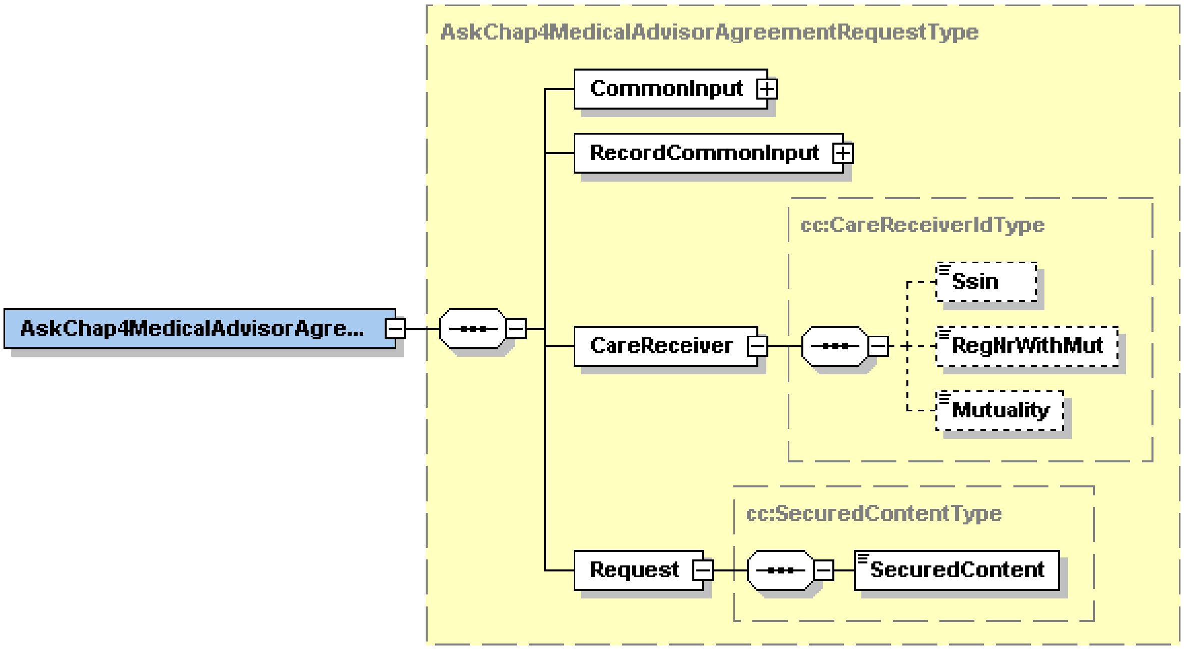

ConsultChap4MedicalAdvisorAgreementResponseType
attributes
core:StatusType
Status
Code
Message
1..∞
ConsultChap4MedicalAdvisorA...
CommonOutput
RecordCommonOutput
cc:FaultType
FaultCode
FaultSource
Contains the name of the subsystem that caused the error event to occur.
ReturnInfo
Message
A human-readable error message.
Detail
0..∞
cc:SecuredContentType
Response
SecuredContent

Field name	Descriptions
Status	The Status element contains a code and a message. If no error has occurred during the call, the Code is set to "200" and the Message is "Success". Otherwise, a soap fault exception is returned (see also Table 1 in section 8 Error and failure messages) or a business error is returned (see ReturnInfo element for more details on the business error).
CommonOutput	See section 5.3.3.3 : CommonOutputType
RecordCommonOutput	See section 5.3.3.4 : RecordCommonOutputType
ReturnInfo	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)
Response	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.2 AskChap4MedicalAdvisorAgreement

5.3.2.1 *Input arguments for AskChap4MedicalAdvisorAgreement request*

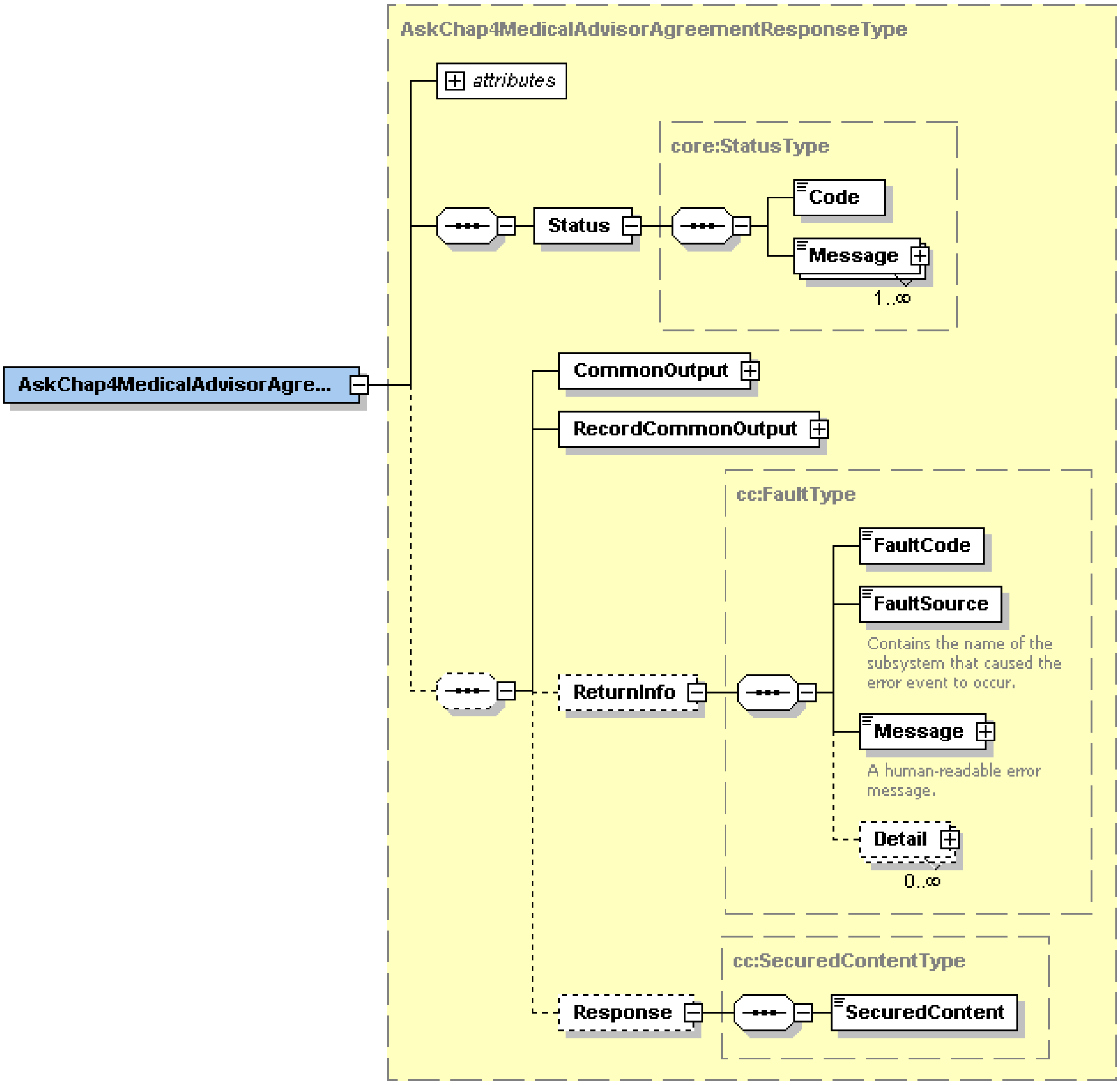
The AskChap4MedicalAdvisorAgreement request has the structure as shown on the below figure:



Field name	Descriptions
CommonInput	See section 5.3.3.1 : CommonInputType
RecordCommonInput	See section 5.3.3.2: RecordCommonInputType
CareReceiver	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)
Request	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.2.2 *AskChap4MedicalAdvisorAgreement response*

The AskChap4MedicalAdvisorAgreementResponse response has the structure as shown on the below figure:

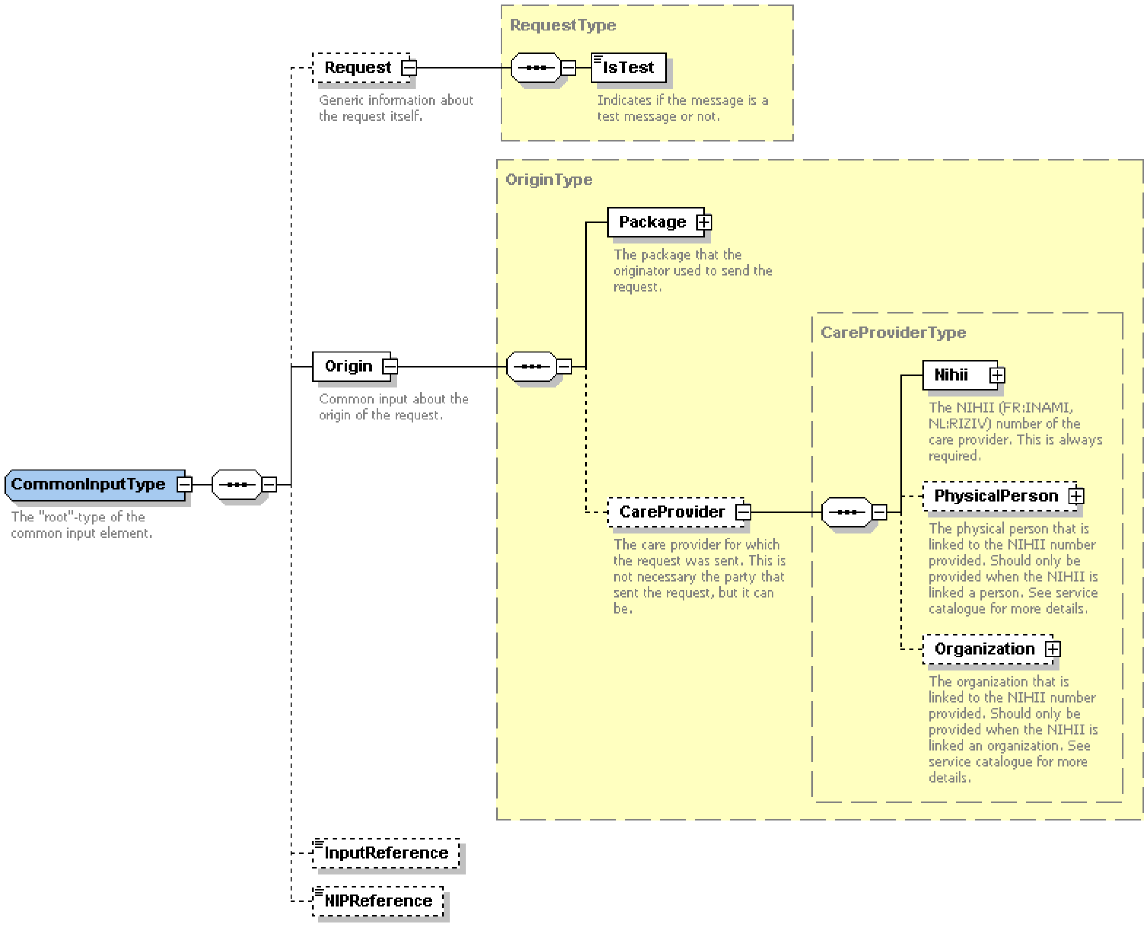


Field name	Descriptions
Status	The Status element contains a code and a message. If no error has occurred during the call, the Code is set to "200" and the Message is "Success". Otherwise, a soap fault exception is returned (see also Table 1 in section 8 Error and failure messages) or a business error is returned (see ReturnInfo element for more details on the business error).
CommonOutput	See section 5.3.3.3 : CommonOutputType
RecordCommonOutput	See section 5.3.3.4 : RecordCommonOutputType
ReturnInfo	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)
Response	See the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.3 Common data structures

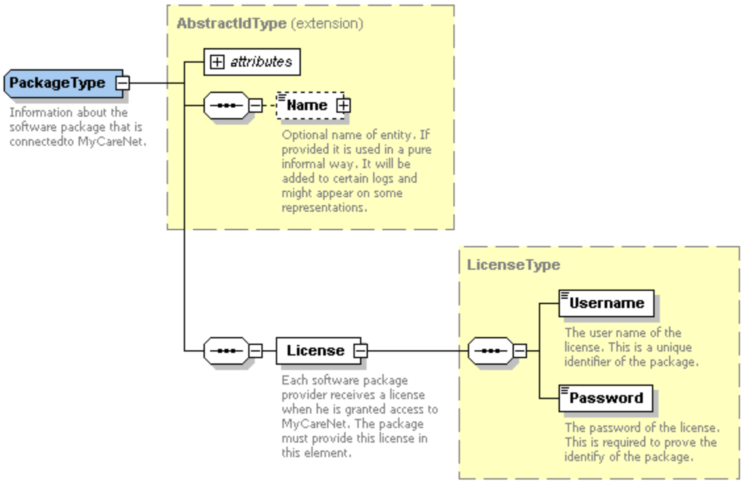
This section provides a description of common data structures.

5.3.3.1 CommonInputType


RequestType
Request
Generic information about the request itself.
IsTest
Indicates if the message is a test message or not.
OriginType
Package
The package that the originator used to send the request.
CareProviderType
Nihii
The NIHII (FR:INAMI, NL:RIZIV) number of the care provider. This is always required.
Origin
Common input about the origin of the request.
CommonInputType
The "root"-type of the common input element.
CareProvider
The care provider for which the request was sent. This is not necessary the party that sent the request, but it can be.
PhysicalPerson
The physical person that is linked to the NIHII number provided. Should only be provided when the NIHII is linked a person. See service catalogue for more details.
Organization
The organization that is linked to the NIHII number provided. Should only be provided when the NIHII is linked an organization. See service catalogue for more details.
InputReference
NIPReference

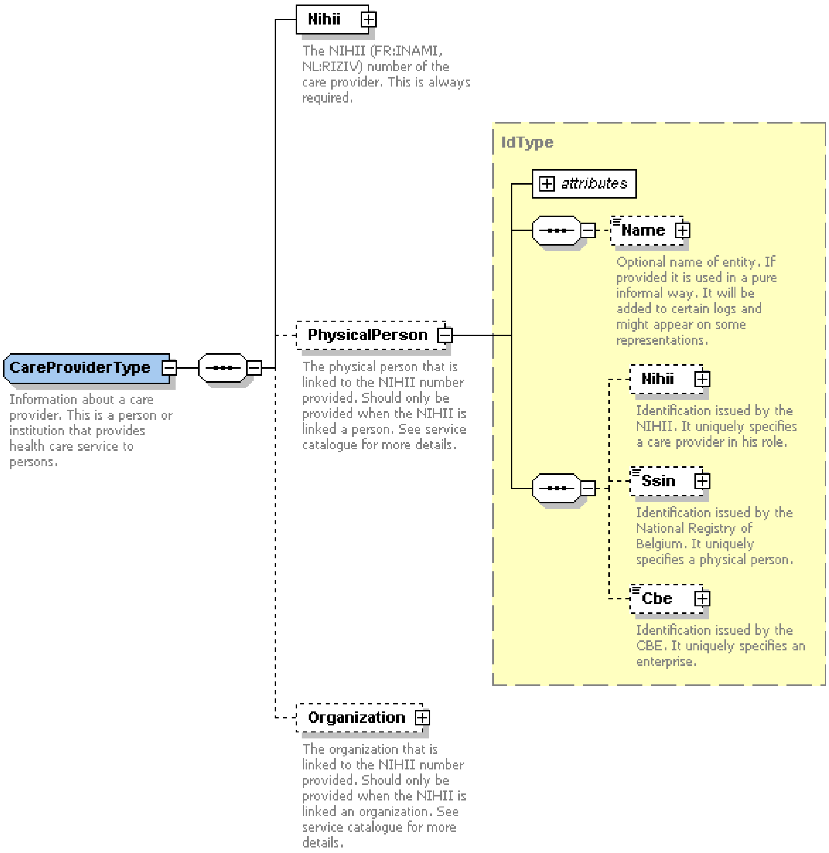
For the semantics of the particular elements and other information about the service, see the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

Package:


AbstractIdType (extension)
attributes
PackageType
Information about the software package that is connectedto MyCareNet.
Name
Optional name of entity. If provided it is used in a pure informal way. It will be added to certain logs and might appear on some representations.
LicenseType
Username
The user name of the license. This is a unique identifier of the package.
License
Each software package provider receives a license when he is granted access to MyCareNet. The package must provide this license in this element.
Password
The password of the license. This is required to prove the identify of the package.

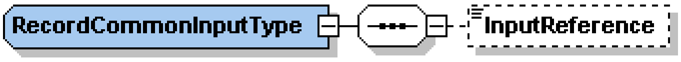
For the semantics of the particular elements and other information about the service, see the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

Care Provider:


Nihii
The NIHII (FR:INAMI, NL:RIZIV) number of the care provider. This is always required.
IdType
attributes
Name
Optional name of entity. If provided it is used in a pure informal way. It will be added to certain logs and might appear on some representations.
PhysicalPerson
The physical person that is linked to the NIHII number provided. Should only be provided when the NIHII is linked a person. See service catalogue for more details.
CareProviderType
Information about a care provider. This is a person or institution that provides health care service to persons.
Nihii
Identification issued by the NIHII. It uniquely specifies a care provider in his role.
Ssin
Identification issued by the National Registry of Belgium. It uniquely specifies a physical person.
Cbe
Identification issued by the CBE. It uniquely specifies an enterprise.
Organization
The organization that is linked to the NIHII number provided. Should only be provided when the NIHII is linked an organization. See service catalogue for more details.

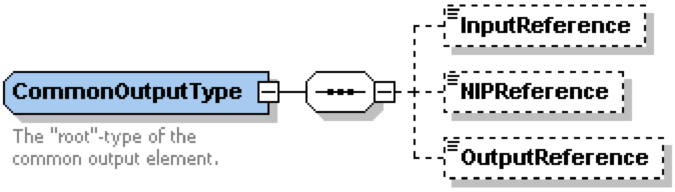
For the semantics of the particular elements, see the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.3.2 RecordCommonInputType



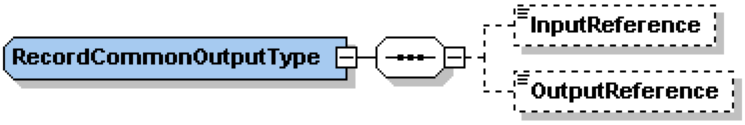
For the semantics of the particular elements, see the documentation ("MyCareNet Service Catalogue", and other) as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.3.3 CommonOutputType



For the semantics of the particular elements, see the as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)

5.3.3.4 RecordCommonOutputType



For the semantics of the particular elements, see the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive).

6. Risks and Security

6.1 Business security

In case the development adds an additional use case based on an existing integration, the eHealth platform has to be informed with a detailed estimation of the expected load, at least one month in advance. This will ensure an effective capacity management.

In case of technical issues on the web service, the partner may obtain support from the contact center responsible for this service.

In case the eHealth platform finds a bug or vulnerability in its software, the partner is advised to update his application with the newest version of the software within 10 business days.

In case the partner finds a bug or vulnerability in the software or WS delivered by the eHealth platform, he is obliged to contact and inform the eHealth platform immediately and he is not allowed to publish this bug or vulnerability in any case.

6.1.1 Web service

WS security used in this manner is in accordance with the common standards. Your call will provide:

- SSL one way
- Time-to-live of the message: one minute. Note that the time-to-live is the time difference between the Created and Expires elements in the Timestamp and is not related to the timeout setting on the eHealth ESB, etc. This means that eHealth will process the message if it is received within the time-to-live value (there is also tolerance of 5 minutes to account for the clock skew), but the actual response time may be greater than one minute in some situations.
- Signature of the timestamp and body. This will allow eHealth to verify the integrity of the message and the identity of the message author.
- Encryption of the business part of the message with the MyCareNet ETK.

6.1.2 The use of username, password and token

The username, password and token are strictly personal. Partners and clients are not allowed to transfer them. Each user takes care of his username, password and token and he is forced to confidentiality of it. Moreover, each user is responsible for any use including the use by a third party, until the inactivation.

7. Test and release procedure

7.1 Procedure

This chapter explains the procedures for testing and releasing an application in acceptation or in production.

7.1.1 Initiation

If you intend to use the eHealth service in the acceptance environment, please contact info@ehealth.fgov.be.

The Project department will provide you with the necessary information and mandatory documents.

7.1.2 Development and test procedure

You have to develop a client in order to connect to our WS. Most of the required integration info is published on the eHealth portal.

In some cases, the eHealth platform provides you with a test case in order for you to test your client before releasing it in the acceptance environment.

7.1.3 Release procedure

When development tests are successful, you can request to access the eHealth acceptance environment.

From this moment, you can start integration and acceptance tests. The eHealth platform suggests testing during a minimum of one month.

After successful acceptance tests, the partner sends his test results and performance results with a sample of "eHealth request" and "eHealth answer" to the eHealth point of contact by email.

Then, the eHealth platform and the partner agree on a release date. The eHealth platform prepares the connection to the production environment and provides the partner with the necessary information. During the release day, the partner gives feedback on the test and performance tests.

For further information and instructions, please contact: integration-support@ehealth.fgov.be.

7.1.4 Operational follow-up

Once in production, the partner using the eHealth service for one of his applications, will always test first in the acceptance environment before releasing any adaptations of his application in production. In addition, he will inform the eHealth platform on the progress and test period.

7.2 Test cases

The eHealth platform recommends performing tests for the following cases:

- Request for agreement to Chapter4 (contact NIC/CIN for test data of the patients)
- Consult patient's agreement to Chapter4 medication (contact NIC/CIN for test data of the patients)

In addition, the software providers should also run negative test cases.

8. Error and failure messages

There are different possible types of response:

- If there are no technical errors, responses as described in the remainder of this section are returned. Section 5 describes the common element types for the responses and the requests. For more details on the specific elements and the concepts behind them, see the documentation as provided by the CIN/NIC (see the documentation contained in the "MCN Chap. IV functional description (fr/nl)" archive)
- In the case of a technical error, a SOAP fault exception is returned (see table 1).

Table 1: Description of the possible SOAP fault exceptions.

Code	Message
SOA-00001	Service error
SOA-01001	Service call not authenticated
SOA-01002	Service call not authorized
SOA-02001	Service temporarily not available. Please try later
SOA-02002	Message must be SOAP
SOA-03001	Malformed message
SOA-03002	Message must be SOAP
SOA-03003	Message must contain SOAP body
SOA-03004	WS-I compliance failure
SOA-03005	WSDL compliance failure
SOA-03006	XSD compliance failure
SOA-03007	Message content validation failure

The soap header (only when the received response is not a SOAP fault) contains a message ID, e.g.:

```
<soapenv:Header>
      <add:MessageID
xmlns:add="http://www.w3.org/2005/08/addressing">6f23cd40-09d2-4d86-b674-
b311f6bdf4a3</add:MessageID>
</soapenv:Header>
```

This message ID is important for tracking of the errors. It should be provided (when available) when requesting support.