Service specification eHealth-Qermid Tuco

This document is provided to you free of charge by

The eHealth platform

Willebroekkaai 38

1000 BRUSSELS

Table of contents

1 Introduction

This document describes how send a request to the Qermid Tuco service. More in particular, it describes the security requirements and the structure of the messages (the interface of the service). Detailed description of the functionality of the service, the semantics of the particular elements and other general information about the service is out of the scope of this document. This kind of information can be found in the documentation provided by Qermid (see the documentation contained in the "Qermid_Tuco web service.zip" archive).

In order to be able to call the Qermid Tuco web service, please follow these steps:

- Obtain a SAML Token from the eHealth SSO service (see section 2).
- Use the eHealth Encryption libraries to encrypt the questionnaire before registration (see section 3).
- Call the webservice:
    - Requests to the Qermid Tuco service are described in section 4.
    - Responses from the service are described in section 5.

If you have technical questions or need more information, you can contact eHealth at info@ehealth.fgov.be.

2 Identification: SSO

This section specifies how to obtain a SAML token from the STS (Secure Token Service) in order to have access to the Qermid Tuco web service. The remainder of this section describes the needed attributes for each type of the user. For more details on how STS works, see https://www.ehealth.fgov.be/fr/support/sts-secure-token-service.

2.1 Cardiologist within a hospital

The SAML token request is secured with the eHealth certificate of the hospital. The certificate used by the Holder-Of-Key verification mechanism is the same eHealth certificate. The needed attributes are the following (AttributeNamespace: "urn:be:fgov:identification-namespace"):

- The social security identification number of the cardiologist: *urn:be:fgov:person:ssin*
- The NIHII number of the hospital: *urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number* and *urn:be:fgov:ehealth:1.0:hospital:nihii-number*

Doctor must also specify which information must be asserted by eHealth:

- The social security identification number of the doctor (AttributeNamespace: "urn:be:fgov:identification-namespace"): *urn:be:fgov:person:ssin*
- The NIHII number of the hospital: *urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number* and *urn:be:fgov:ehealth:1.0:hospital:nihii-number*
- The NIHII number of the doctor (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*): *urn:be:fgov:person:ssin:ehealth:1.0:doctor:nihii11*
- To have access to the Qermid Tuco web service, the person must be a recognized doctor (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*) *urn:be:fgov:person:ssin:doctor:boolean*
- To have access to the Qermid Tuco web service, the hospital must be a recognized hospital (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*) *urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number:recognisedhospital:boolean*

3 Encrypted message

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 (https://www.ehealth.fgov.be/fr/support/services-de-base/systeme-de-cryptage-end-to-end).

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

Environment	Type	Value	Application ID
Integration Test Environment	CBE	0206653946	ECAREACC
Production Environment	CBE	0206653946	ECAREPRD

More information can be found in the documentation provided by Qermid (see the documentation contained in the "Qermid_Tuco web service.zip" archive).

4 Request to the Qermid Tuco webservice

To call the Qermid Tuco webservice:

- 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 (holder-of-key). (link: http://www.oasis-open.org/committees/download.php/16768/wssv1.1-spec-os-SAMLTokenProfile.pdf).
    - **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[1].

As for now, only the operations described below are available. The operations for the web services are:

- SendEcareTucoDeclaration
- UpdateEcareTucoDeclaration (not yet available)
- DeleteEcareTucoDeclaration
- ConsultEcareTucoDeclaration (not yet available)

Web service targeted	Environment	Url
SendEcareTucoDeclaration	Integration Test Environment	https://services-acpt.ehealth.fgov.be/QermidTuCo/Send/v1?WSDL
	Production Environment	https://services.ehealth.fgov.be/QermidTuCo/Send/v1?WSDL
UpdateEcareTucoDeclaration	Integration Test Environment	https://services-acpt.ehealth.fgov.be/QermidTuCo/Update/v1?WSDL
	Production Environment	https://services.ehealth.fgov.be/QermidTuCo/Update/v1?WSDL
DeleteEcareTucoDeclaration	Integration Test Environment	https://services-acpt.ehealth.fgov.be/QermidTuCo/Delete/v1?WSDL
	Production Environment	https://services.ehealth.fgov.be/QermidTuCo/Delete/v1?WSDL
ConsultEcareTucoDeclaration	Integration Test	https://services-acpt.ehealth.fgov.be/QermidTuCo/Consult/v1?WSDL

[1] WSDL's can be found in the eHealth Service Registry: https://services.ehealth.fgov.be/registry/uddi/bsc/web

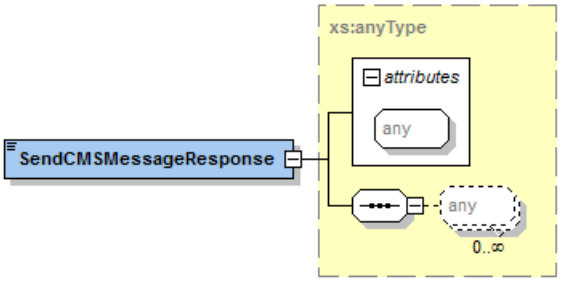
	Environment	
	Production Environment	https://services.ehealth.fgov.be/QermidTuCo/Consult/v1?WSDL

The remainder of this section describes the structure of the input and output request messages. The response messages are described in Section 5. For more detail, see the documentation as provided by Qermid (see the documentation contained in the "Qermid_Tuco web service.zip" archive)

4.1 Input request

The input request is defined by a tag which will contain the encrypted request. For more detail, see the documentation as provided by Qermid (see the documentation contained in the "Qermid_Tuco web service.zip" archive)

4.2 Output Response



The output request is defined by a tag which will contain the encrypted response provided by the Qermid Tuco web service. For more detail, see the documentation as provided by Qermid (see the documentation contained in the "Qermid_Tuco web service.zip" archive)

5 Response from the Qermid Tuco webservice

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 detail on the specific elements and the concepts behind them, see the documentation as provided by Qermid (see the documentation contained in the "Qermid_Tuco web services.zip" 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.