Logius
Ministerie van Binnenlandse Zaken en Koninkrijksrelaties

• Digipoort Interface description WUS 2.0 Companies

Interface version 1.2

Version	1.0
Date	30 May 2012
Status	Final

Publisher's imprint

Project name	Digipoort
Version number	1.0
Organisation	Logius P.O. Box 96810 2509 JE The Hague servicecentrum@logius.nl

Appendix/appendices Service description Supply Service
Service description Status Information Service
Service description Delivery Service Service description Retrieve Service
Digipoort error messages and status notifications

Content

Introduction

Objective and target group

This document describes the agreements made with regard to electronic messaging within the government through Digipoort (previously the Government Gateway).

This document is intended for developers of software that supplies and requests messages to and from the government through this infrastructure.

Outline of the report

This interface description forms the basis of a series of service descriptions that provide an insight into the use of the Digipoort services. This document has been compiled as follows:

- The first chapter contains general information about the operation of Digipoort;
- The second chapter gives a broad description of the operation of the "WUS 2.0 for Companies" interface and the web services that are involved.
- The third chapter provides a broad description of the SOAP message;
- The fourth and fifth chapters describe the definitions of the various protocols;
- The sixth chapter provides an overview of all generally applicable standards and agreements.

This interface description forms part of a larger set of documents that describe the Digipoort services.

Status

This document describes the agreements made with regard to Digipoort's "WUS 2.0 for Companies" interface. Expectations are that the open standards that are used will continue to develop in future years and that the communication need will also be subject to change. The consequence of this is that, during future years, there will be new releases of Digipoort. That can have an effect on the interface. Logius is aiming to develop new releases in close consultation with the market. To enable market parties to quickly and easily use Digipoort, a decision has been made to use open standards and existing tools as far as possible. Examples of that are the use of the SOAP protocol and the application of PKIoverheid certificates.

Assistance

Information relating to assistance with the use of Digipoort services is available on the website: www.logius.nl/producten/gegevensuitwisseling/digipoort.

1 Electronic messaging

1.1 Introduction

Digipoort offers services targeted at companies and services targeted at the government. There are also services that support the implementation of the handling processes, such as the authorisation service and the validation service.

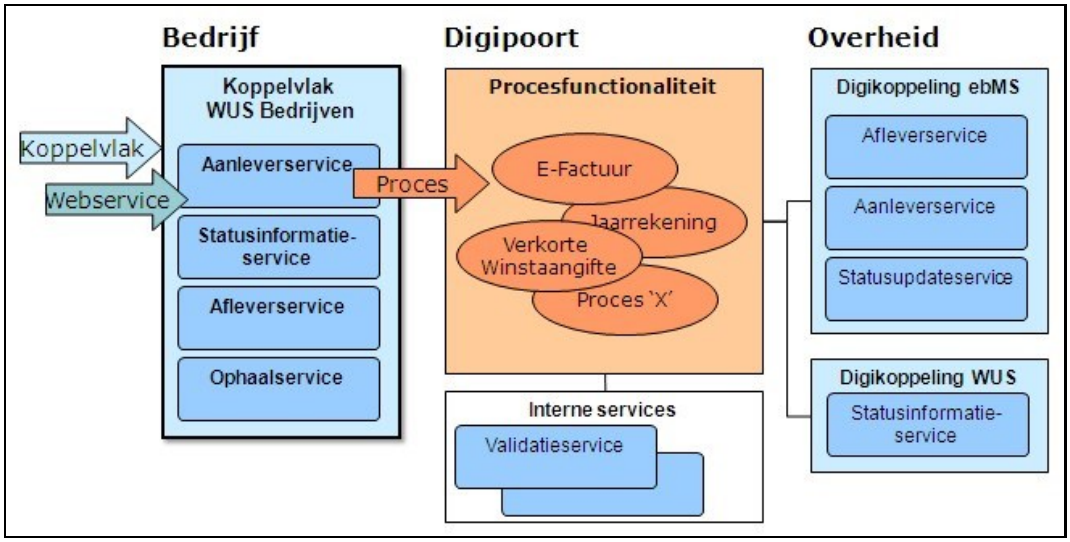

Figure 1: Services within Digipoort

The services are shown in the diagram below.

This interface description forms the basis of the services offered by Digipoort to companies. These services include the Aanleverservice (Supply Service),

Statusinformatieservice (Status Information Service), Afleverservice (Delivery Service) and Ophaalservice (Retrieve Service). All services are also actually offered as web services on Digipoort, except for the Afleverservice (Delivery Service). As the foregoing concerns messages that are sent *by* Digipoort *to the company*, this web service has to be implemented by the company itself.

The details of each service are defined in separate documents: the Service Descriptions.

The interface can be expanded with new services. These will then always comply with this interface description.

1.2 Security

1.2.1 Level of transport

It has to be possible for all participating parties to establish the authenticity of systems in Digipoort and of the users of a service before a data communication session is started. The authenticity of systems is checked using PKIoverheid certificates[1].

For a production connection to Digipoort, you have to use a PKIoverheid certificate (X.509). This certificate guarantees the security and reliability of the connection between your system and Digipoort. Please note, the PKIoverheid certificate is only compulsory within the production environment. In the pre-production environment, self-signed test certificates can also be used, supplied by Logius. You can request a PKIoverheid certificate from a Certificate Service Provider (CSP). An overview of the current CSPs can be found on the Logius website(http://www.logius.nl/producten/toegang/pkioverheid/aansluiten/toetreden-tot-pkioverheid/, under Toegetreden CSPs). Because of the lead-time of the application, we recommend that you apply for a PKIoverheid certificate in good time. For a test certificate, you can apply to the Logius Service Centre (servicecentrum@logius.nl).

In fact, the authenticity of companies is determined on the basis of the PKIoverheid client certificate that is located on the client system. Using this certificate, the client opens a connection in accordance with the TLS/SSL protocol (see the overview in figure 2). In addition to authentication, this protocol also offers encryption at transport level.

The validity of the client certificate is verified based on the data contained in the certificate. In addition, using a Certificate Revocation List (CRL) it is checked that the certificate has not been withdrawn.

[1] *In non-production environments, test certificates are used.*

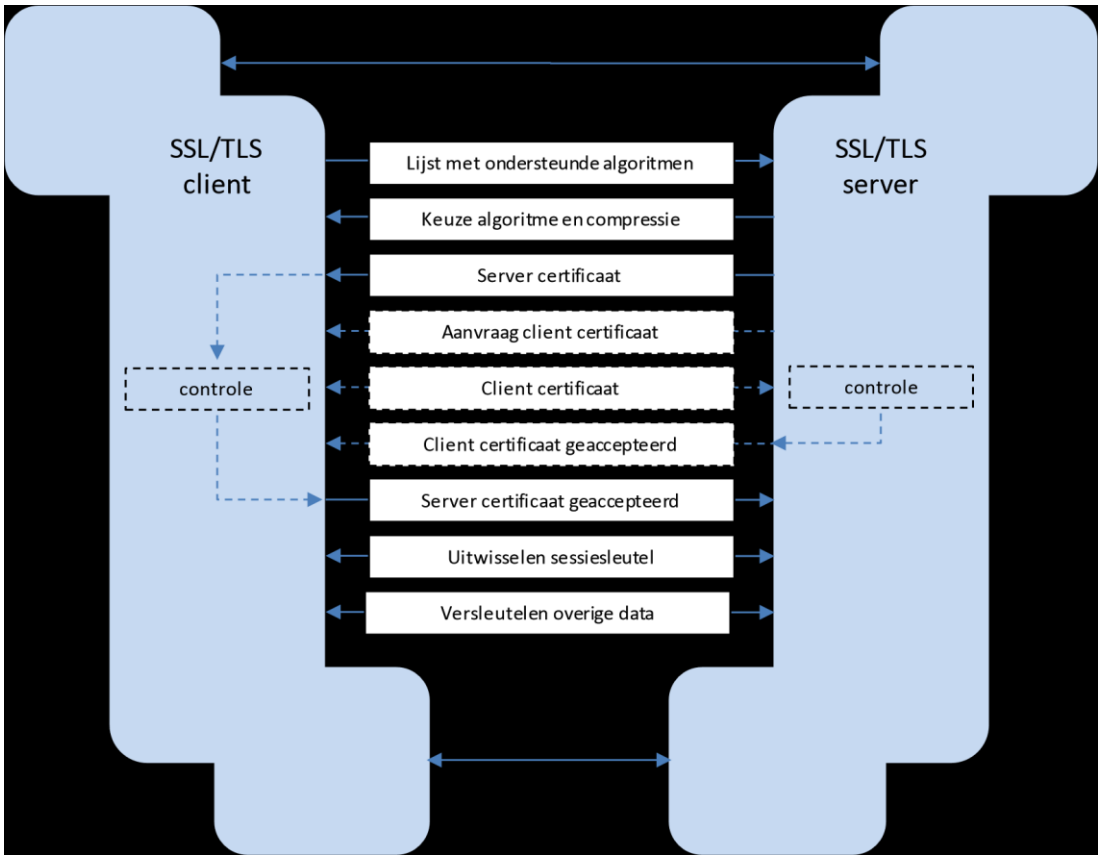

Figure 2: TLS/SSL Communication

At transportation level, the party to undergo authentication is the party with which the TLS connection is established. This can also be an intermediary that arranges the connection with Digipoort for one or more companies. At transportation level, it is therefore not necessarily the "owner" of the messages (the company on behalf of which the invoice, etc. is sent) whose identity is checked.

1.2.2 Message level

At message level, security is administered through the use of WS Security. The message has to be secured with a signature across the SOAP body and the SOAP header elements. The certificate that is used for this has to comply with the same requirements as the certificate that is used at transport level. However, it does not have to be the same certificate.

This security guarantees the integrity of the message itself. Also, if the message is archived, the WS-Security information is saved with the message.

Verification of the WS-Security signature means that the signature is placed using a valid certificate and that there is a relationship between the certificate and the company to which the message relates. This relationship might be that the certificate belongs to the company itself, or that the certificate belongs to a party that is authorised by the company to exchange information on behalf of the company with public authorities.

The verification of the identity, that is represented by the certificate, and the authorisation of the relevant party is done later on during the processing. During the application, only the legitimacy of the certificate and of the signature are verified. The company number and message type have to be available in the supply request, to enable the later authorisation, therefore these are also used for verification.

2 Course of the session

A web service client of a company effects a TLS connection with a Digipoort web service, or the other way around. SOAP request messages are sent over this connection (for more information about the structure of the SOAP messages, see chapter 3).

If the message does not fulfil the requirements laid down in the WSDL, a SOAP fault is returned. If the message does fulfil the requirements, it will undergo further processing. In the event that the message cannot be processed correctly, a SOAP fault will be returned. If the message processed successfully, a SOAP response is sent.

Each service comprises at least the following parts:

- Monitoring the request
- Receipt (of the monitored) request
- Sending a response

In addition to the aforementioned elements, for each service other elements can be included. These are detailed in the description of the service.

2.1 Monitoring the request

SOAP messages that are presented to Digipoort and SOAP messages that are sent by Digipoort to a company, are prepared in accordance with a pre-defined structure (SOAP request). This structure is recorded in an XML Diagram (XSD), that, in turn, is incorporated in the WSDL that formally describes the web service. WSDL and XSD are included in the description of each service.

After a request (in the form of a SOAP message) has been received by Digipoort or by the company, the following aspects have to be checked:

Verification	**Clarification**
Is an element present?	This is verifying whether all compulsory elements, as described in the WSDL, are included in the supply request.

Is no unknown element present?	This is verifying whether any elements are included in the requests that are not described in the WSDL.
Does the element contain a value?	This is verifying whether all compulsory elements also actually contain a value.
Does it relate to a permitted value?	This is verifying whether all elements contain permitted values.
Is the length of the value correct?	This is verifying that the value of the elements does not exceed the length described in the WSDL.

2.2 Receipt of a request

Every request for a Digipoort service is recorded in the message administration system. The message administration system acts internally within Digipoort as an audit trail. In the same way, the company can record requests from Digipoort in its own message administration system.

2.3 Sending an answer

The answer is sent when the request fulfils all stipulated requirements.

Each answer to Digipoort is recorded in the message administration system. The company can also record answers from Digipoort in its own message administration system.

The elements of the answer are described in the service description of the relevant service.

3 SOAP message

The "WUS 2.0 for Companies" interface uses the SOAP 1.1 standard for composing electronic messages. SOAP is a common standard for electronic messaging based on services.

A message that is sent to a service is called a "SOAP request". A "SOAP response" can be returned as a response to a request. If errors are found upon receipt of or whilst processing the request message, a "SOAP fault" is returned, in which further information is provided about the error that has been found. A description of the error message is recorded in the set of documentation.

3.1 Structure

The structure of request and response messages depends on the service within which these messages are used. A detailed description can therefore be found in the separate service descriptions.

Under interface version 1.2, the services use a generic diagram (XSD), in which all message types are specified. This XSD is appended separately in the set of documentation.

3.2 Addressing

Digipoort services under the "WUS 2.0 for Companies" interface use WS-Addressing, which enables messages to be routed irrespective of the transport protocol that is used.

More details about WS-Addressing can be found in chapter 4.

3.3 Signing a message (WS-Security)

Messages should be signed digitally. Use the WS-Security standard to sign messages. Signing applies to both request and response messages.

More information about the application of this can be found in chapter 5.

3.4 MTOM

The data contained in the message are recorded in the message content element. It is also possible to record additional attachments.

Attachments can be recorded in the message in two ways:

- As Base64 coded binary data;
- Based on MTOM.

When using MTOM, this is also sometimes referred to as an optimised message. MTOM is described in WS-I Basic Profile 1.2 (see http://www.w3.org/TR/soap12-mtom/)

The most common toolkits can receive and send MTOM messages. As a rule, it can be stipulated whether or not MTOM should be used by means of a configuration file or through the code. This is how the web service can be informed whether this MTOM is used or can be used when receiving and sending messages. The actual use of MTOM is actually specified by the service requester; the service requester will take the initiative in this respect. If a web service set up according to MTOM receives an optimised message, an optimised response will also be returned. If the request was not optimised (MTOM is not used), neither will the response be optimised.

4 WS Addressing

Digipoort uses WS Addressing 1.0 with name space http://www.w3.org/2005/08/addressing.

The WS Addressing elements of the SOAP requests and responses have to be adopted as follows:

Element	**Clarification**	**Compulsory**
wsa:To	The WSDL value (request) or http://www.w3.org/2005/08/addressing/none or http://www.w3.org/2005/08/addressing/anonymous (response)	Yes
wsa:Action	This value is used to invoke a specific operation.	Yes
wsa:MessageID	The unique ID for this message as UUID.	Yes
Wsa:RelatesTo	Contains the value of the wsa:MessageID of the original request.	Only in response messages
wsa:ReplyTo	http://www.w3.org/2005/08/addressing/anonymous	No

5 WS Security

A company or intermediary has to sign parts of a SOAP message. These parts are, as it were, furnished with an electronic signature, by applying an encryption using a PKIOverheid certificate.

The certificate, the signature and the algorithms that are used have to be included as a WS Security element in the message header.

Example (signature of the Body):

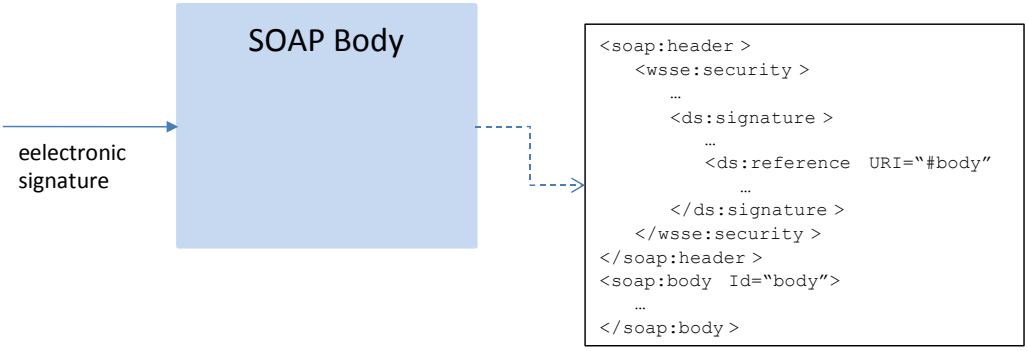

Figure 3 Digital signature under WS Security

- The use of WS Security enables the following:
- The ability to check the integrity of the message;
- The guarantee of the identity of the sender of the message;
- Inclusion of a time stamp in the message, which indicates when the message was created and (optionally) until when it can be processed. Among others, this prevents there being an attack on Digipoort.

The public key of the certificate by which the signature is placed, has to be included in the header of the SOAP envelope as binary security token.

5.1 Signing the message

The following parts are signed:

- soap-env:Body
- the header part Time stamp
- the header part WS Addressing (all elements)

The following requirements apply for the WS Security elements:
http://www.w3.org/2000/09/xmldsig#

Step 1: Canonicalization http://www.w3.org/2001/10/xml-exc-c14n#

Step 2: Digest http://www.w3.org/2000/09/xmldsig#sha1

Step 3: Signature http://www.w3.org/2000/09/xmldsig#rsa-sha1

5.2 Time stamp Created

Within the "TimeStamp" element, the "Created" element provides the date and the time at which the request is sent from or to Digipoort. The time stamp is expected in the UTC form (Zulu Time) in line with the format below. In addition, the "Expires" option allows indication of which period of time the message has to be dealt with.

Example:

```
<wsu:Timestamp ... >
       <wsu:Created>2011-11-30T11:12:12.459Z</wsu:Created>
       <wsu:Expires>2011-12-01T11:12:12.459Z</wsu:Expires>
</wsu:Timestamp>
```

These WS Security header elements belong in the web service utility namespace: http://docs.oasis-open.org/wss/2004/01/oasis-200401-wsshttp://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsdwssecurity-utility-1.0.xsd

Element	TimeStamp
Compulsory	Yes

Element	Created
Compulsory	Yes
Type	DateTime in UTC

Element	Expires

Compulsory	No
Type	DateTime in UTC

6 General agreements

6.1 Communication standards

The communication between the web service client and the web service covers a number of layers. Standards applicable for each level. In summary, this involves the following standards:

Level	**Standard**
Application layer	XML
	SOAP
Session layer	HTTP
Transport layer	TCP
Network layer	IP

6.2 Prefixes

For namespaces in the wsdl and SOAP messages of the services, the prefixes listed below are used:

Prefix	**Specification**	**Namespace URI**
Tns	WUS 2.0 <Digipoort_Service> 1.0	http://logius.nl/digipoort/wus/2.0/<Digipoort_Service>/1.0/ for every Digipoort WUS service
soapenv	SOAP 1.1	http://schemas.xmlsoap.org/soap/envelope/

Wsdl	WSDL 1.1	http://schemas.xmlsoap.org/wsdl
Ds	XML Signature 1.0	http://www.w3.org/2000/09/xmldsig#
Xsd	XML Diagram 1.0	http://www.w3.org/2001/XMLSchema
wsse	WS Security 1.0	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
Wsu	WS Security 1.0	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd
Wsa	WS Addressing 1.0	http://www.w3.org/2005/08/addressing
Wsam	WS-Addressing 1.0 - Metadata	http://www.w3.org/2007/05/addressing/metadata
Wsp	Webservices Policy 1.2	http://schemas.xmlsoap.org/ws/2004/09/policy
Sp	Security Policy 1.1	http://schemas.xmlsoap.org/ws/2005/07/securitypolicy

6.3 Character coding and character set

The supporting character set is UTF-8.

6.4 Date and time

For all date/time fields, the type xsd:date and xsd:dateTime are used, completed according to the UTC (Z) version on the ISO 8601 (NEN28601) standard. The use of fractions of seconds is optional.

6.5 Standards that are used

For these standards, the same choices are made as for the Digikoppeling standard WUS 2.0 that applies to the Dutch government. See http://www.logius.nl/producten/gegevensuitwisseling/digikoppeling/.

Government standards:

- PKI overheid 1.1

WS-I standards:

- WS-I Basic Profile 1.2
- WS-I Basic Security Profile 1.0

W3C standard:

- MTOM 1.0