



Logius
*Ministerie van Binnenlandse Zaken en
Koninkrijksrelaties*

Interface Description Digipoort Message exchange - SMTP-MTA (server-to-server)

Version 2.1.2

Date	1 June 2015
Status	Definitive

Publisher's imprint

Project name	Digipoort
Version number	2.1.2
Organization	Logius Postbus 96810 2509 JE Den Haag servicecentrum@logius.nl
Appendix	0

Content

Publisher's imprint.....	2
Content	3
Introduction.....	4
<i>Objective and Target Group.....</i>	<i>4</i>
<i>Outline of the report</i>	<i>4</i>
<i>Status.....</i>	<i>4</i>
1 Interaction through the interface.....	5
1.1 <i>Transport</i>	<i>5</i>
1.2 <i>Contents.....</i>	<i>5</i>
1.2.1 SMTP Envelope	5
1.2.2 SMTP Headers	6
1.2.3 SMTP Body	6
1.2.4 Delivery Status Notifications (DSNs)	8
1.2.5 Addressing.....	8
1.3 <i>Security</i>	<i>8</i>
1.3.1 Non-repudiation	8
1.3.2 Authenticity.....	8
1.3.3 Confidentiality.....	8
1.3.4 Integrity	8
1.4 <i>Examples.....</i>	<i>8</i>
1.4.1 Incoming SMTP message.....	9
1.4.2 Outgoing SMTP message	9
1.4.3 Delivery report.....	10
2 General arrangements.....	12
2.1 <i>Standards</i>	<i>12</i>
2.1.1 Normative RFCs for electronic messaging	12
2.1.2 Normative RFCs for message transport.....	12
2.2 <i>Preconditions</i>	<i>12</i>
2.3 <i>Error messages.....</i>	<i>12</i>
2.4 <i>Addresses</i>	<i>13</i>
2.5 <i>Limits.....</i>	<i>13</i>
2.6 <i>Support.....</i>	<i>13</i>

Introduction

Objective and Target Group

The aim of Digipoort (formerly the Government Gateway OTP) is to enable a generic electronic access service through which the business community can reach the entire government.

Whether or not Digipoort will function successfully is very dependent on the proper description of the interfaces to which the government, the business community and citizens have to (are able to) connect.

Digipoort offers the business community and the government various interfaces. A separate specification is available for each interface. This document sets out one of these interfaces, i.e. the SMTP (Simple Mail Transfer Protocol) interface for the exchange of electronic messages between mail servers (MTAs). This interface specification does not describe the exchange of electronic messages between a mail client and Digipoort. Separate interface descriptions are available for this: SMTP-MSA and POP3.

This document is primarily intended for developers of system-to-system connections.

Outline of the report

The structure of the document is as follows. The first chapter contains general information. The second chapter contains the description of the functioning of the delivery. The third chapter provides a more detailed insight into the technical functioning of the interface. The document closes with an overview of all generally applicable standards and rules.

Status

This document is a version of the Digipoort SMTP-MTA interface specifications version 2.1, which has been added to and rewritten, for the migration of the Government Gateway.

This document describes the current situation in terms of the interface.

The operation of the interface has not changed in respect of version 2.1.

1 Interaction through the interface

1.1 Transport

A connection can be made with Digipoort in the following ways:

1. Internet
 - a. VPN using IPsec
2. Diginetwork (previously Coupling Network for the Public Sector; only for public authorities)

For this interface, a connection via SSL/TLS over TCP/IP is not possible. This is possible for the SMTP-MSA and POP3 interfaces.

The SMTP protocol is a generally used message transport protocol. In this document, this protocol will be broadly outlined; the emphasis will be on describing the exceptions and/or restrictions in respect of the applicable standards. See paragraph 2.1 for an overview of the standards

For SMTP-MTA, it is a requirement that the party sending the message has a fully functioning SMTP server: an SMTP server that can send and receive SMTP messages and that supports the RFCs that are listed in paragraph 2.1.

1.2 Contents

Digipoort offers a basic service, which can be used by businesses and public authorities. The SMTP interface has been set up in such a way that this offers a platform for sending messages to public authorities. The interface specification indicates which specifications are used for sending messages and which standards are used to construct messages.

In this interface specification, the actual content of the message that is sent through SMTP is not mentioned. These are specific chain agreements, that have to be described in the definitions that are prepared by the various chain partners.

An SMTP message that is exchanged between mail servers (MTAs) is compiled from the following parts:

- SMTP Envelope
- The actual message that is compiled from an SMTP Header and from an SMTP Body

Below a further explanation is given about the various parts. For details, we refer to the RFCs that are outlined in paragraph 2.1

1.2.1 SMTP Envelope

The envelope contains the information that is needed to route SMTP traffic between mail servers (MTAs). Digipoort only routes messages based on the information in the envelope.

The envelope contains the following elements (verbs):

Element	Description
---------	-------------

MAIL FROM <originator address> {ENVID <envelope ID>}	The sender of the mail (originator address). Every mail has exactly 1 sender. If a receipt confirmation is requested, a unique ID has to be given (envelope id)
RCPT TO <recipient address> NOTIFY SUCCESS, FAILURE	The recipient of the mail (recipient address). Each mail has exactly 1 sender. Using the NOTIFY option, it can be stipulated that receipt confirmations (or messages indicating no receipt) are sent by the recipient. See 1.2.4 for a further explanation about the receipt confirmations.
DATA	The message content (SMTP headers and body).

1.2.2

SMTP Headers

The headers contain information about the message content: who has sent the message, for whom the message is intended, the subject of the message, etc. The following elements are found (only the ones that are most commonly used are mentioned):

Element	Explanation
from	Address of the sender of the message
to	Address of the recipient of the message.
message ID	A unique identification of the message. The sender can complete this, provided that this is unique. If there is no message ID, Digipoort will add its own message ID.
subject	A description of the subject of the message. This element is given to the recipient of the message.
date	The transmission date of the message provided by the sender.

Header elements mentioned in the RFC that can occur in the SMTP header, but that are not named above, are not copied over by Digipoort. Digipoort also translates the address headers from/to logical Digipoort addresses. Digipoort assigns its own message identifier to each outgoing message.

1.2.3

SMTP Body

Digipoort can receive messages as single part MIME, or as multipart MIME.

Digipoort always sends messages through as single part.

Primarily, the SMTP protocol is intended for the exchange of messages between people. In respect of the Digipoort interface, messages are exchanged between applications (system-system linkage). This means

that, with regard to the MIME RFCs, there are a number of restrictions. These restrictions are required to enable message conversion from and to other interfaces:

When a multipart MIME is used, the content-type is always 'multipart/mixed' and the following restrictions apply:

- There is either exactly one part available, which includes the same information as a single part
- or there are exactly two parts, the first of which is content-type text/plain and is otherwise empty, and the second contains the same information as a single part

There is therefore always exactly one meaningful part present. The following content parameters apply to this part:

Parameter	Value	Explanation
Content-Type	text/plain; charset=us-ascii	Preferential value for text-based messages. Other values are permitted for charset ¹ . If there is no value, us-ascii is assumed. Windows Code Pages (CPxxxx) may not be referenced. These cannot be interpreted on non-Windows systems.
	application/edifact	alternative for messages in edifact format.
	application/octet-stream	application/octet-stream has to be used to transmit binary files.
Content-Transfer-Encoding	base64	preference value
	quoted-printable	alternative for base64;
	7bit	Permitted, but is advised against. Is unsuitable for binary data and text with ascii values in excess of 127. CR/LF details can be lost upon conversion to base64. If the parameter Content-Transfer-Encoding is absent or the value is not entered, this is always interpreted as '7bit'.
Content-Disposition	attachment Filename=<filename>. <extension>	Digipoort always replaces the filename with a unique ID. If the parameter is absent or the value is not entered, Digipoort generates a unique filename with extension .txt. NB: A 'name' can also be entered for the parameter Content-Type. Use of that field is advised against in accordance with RFC 2183.

¹ In any case, the following character sets are supported: "us-ascii", "UTF-8", "ISO-8859-1" (Latin1, West European), "ISO-8859-2" (Latin2, East European), "ISO-8859-5" (Cyrillic), "ISO-8859-6" (Arabic), "ISO-8859-7" (Greek), "ISO-8859-8" (Hebrew), "ISO-8859-9" (Latin5, Turkish), "ISO-8859-10" (Latin6, Lappish/Nordic/Eskimo), "ISO-8859-13" (Latin7, Baltic Rim), "ISO-8859-14" (Latin8, Celtic), "ISO-8859-15" (Latin9, saner West european + euro), "ISO-8859-16" (Latin10, South-Eastern European)

1.2.4 *Delivery Status Notifications (DSNs)*

The forwarding party can receive a DSN message in return (in accordance with RFC 1891) which will state that the message has or has not been correctly delivered

Using the NOTIFY option, the recipient can be asked to send back a DSN that shows the status of receipt of the message.

Success and failure are shown respectively with the parameter values NOTIFY = SUCCESS and NOTIFY = FAILURE.

For all parties, Digipoort enforces the use of the SUCCESS and FAILURE as notify options.

1.2.5 *Addressing*

Digipoort routes based on logical addresses. In other words, that a company never sends a message directly to a government party, but instead of that sends the company a message to an address in the Digipoort domain (and the other way around). Even if Digipoort sends a message, the actual address of the sender never appears in that message, but a logical Digipoort address.

A logical address is used in Digipoort as a one-to-one representation of a physical address (translation rule).

1.3 **Security**

Sessions are secured at the level of the connection.

1.3.1 *Non-repudiation*

Digipoort always establishes which path a message has taken from sender to recipient, along with the time of processing. Digipoort also establishes that the server of the recipient (business or government) has successfully received a message ('SMTP OK').

1.3.2 *Authenticity*

Authentication of the sender and recipient are dealt with for this interface at transport level.

1.3.3 *Confidentiality*

Message encryption is not set up at interface level. As Digipoort has nothing to do with the content of a message, payload encryption can be agreed between the flow owner and supplying parties. This does raise questions regarding the reusability of data: if messages are encrypted for government party X, these cannot be freely reused by government party Y.

1.3.4 *Integrity*

Digipoort does not view the content of the message, other than in the form of a technical integrity check.

1.4 **Examples**

The examples included here show an incoming, an outgoing, and a positive delivery report. The SMTP headers are shown in blue, MIME headers in green and the body in red.

1.4.1 Incoming SMTP message

The example below shows an incoming message containing a multipart MIME with two body parts: the first is empty and the second contains a text/plain) body part.

```
Date: Thu, 27 Aug 2009 13:29:57 +0200 (West-Europa (zomertijd))
From: <opt@nl.bedrijf.com>
To: <def51026@mail.otpnet.nl>
Subject: EDI-Message
MIME-Version: 1.0
Message-Id: <0908271329580A.02960@dna370.w2k.bedrijf.com>
Content-Type: Multipart/Mixed;
    boundary="Boundary-=_vLAGzZmiiwgcHANSqmIDnfGRqrQc"

--Boundary-=_vLAGzZmiiwgcHANSqmIDnfGRqrQc
Content-Type: Text/Plain
Content-Transfer-Encoding: 7bit

--Boundary-=_vLAGzZmiiwgcHANSqmIDnfGRqrQc
Content-Type: Text/Plain
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="IE040601.INT"

TG9yZW0gaXBzdW0gZG9sb3Igc2l0IGFtZXQsIGNvbml1Y3RldHVyIGFkaXBpc2ljaW5nIG
VsaXQs
IHNLZCBkbyBlaXVzbW9kIHRlbXBvcilBpbmNpZG1kdW50IHV0IGxhYm9yZSBldCBkb2xvcm
UgbWFn
bmEgYWxpZXVhLiBVdCB1bmltIGFkIGlpbmltIHZ1bmlhbSwgcXVpcyBub3N0cnVklGV4ZX
JjaXRh
dGlvbiB1bGxhbWVvIGxhYm9yaXMgbmlzaSBldCBhbGlxdWlwIGV4IGVhIGNvbWlwZG8gY2
9uc2Vx
dWF0LiBEdWlzIGF1dGUgaXJ1cmUgZG9sb3IgaW4gcmlvcmVoZW5kZXJpdCBpb2B2x1cH
RhdGUg
dmVsaXQgZG9sb3IgaW4gcmlvcmVoZW5kZXJpdCBudWxsYSBwYXJpYXR1ci4gRX
hjZXB0
ZXVvIHNPbnQgb2NjYWVjYXQgY3VwaWRhZGF0IG5vbiBwcm9pZGVudCwgc3VudCBpb2B2dW
xwYSBx
dWkgb2ZmaWVvYXNkZm9udCBudWxsYSBwYXJpYXR1ci4gRX
dWkgb2ZmaWVvYXNkZm9udCBudWxsYSBwYXJpYXR1ci4gRX

--Boundary-=_vLAGzZmiiwgcHANSqmIDnfGRqrQc--
```

1.4.2 Outgoing SMTP message

The example below shows an outgoing message which originates from Customs. Digipoort sends this as a single part/mime message

```
X-Env-Sender: dinv1234@mail.otpnet.nl
X-Env-Recipient: sagittainvoer@nl.defg.net NOTIFY=10
X-DSN-RET: 2
X-DSN-ENVID: P1B61H10314668A1
X-End-of-Envelope:
X-Mailer: OtpTransfer v1.083
Message-ID: <2009082716212002*/OU=DOUANE-INV-
PROD/O=CENTRAAL/PRMD=BELASTINGDIENST/ADMD=400NET/C=NL/@MHS>
From: <dinv1234@mail.otpnet.nl>
To: <sagittainvoer@nl.defg.net>
Subject: //2009082716212002/1GT4H700/
Date: Thu, 27 Aug 2009 16:21:20 +0100
MIME-Version: 1.0
Content-Type: text/plain; charset=US-ASCII; name="dinv1020-20090827-
142122.164-P1B61H10314668A1.txt"
Content-Disposition: attachment; filename="dinv1020-20090827-
142122.164-P1B61H10314668A1.txt"
Content-Transfer-Encoding: base64

TG9yZW0gaXBzdW0gZG9sb3Igc2l0IGFtZXQsIGNvbml1Y3RldHVyIGFkaXBpc2ljaW5nIG
VsaXQs
IHN1ZCBkbyBlaXVzbW9kIHRlbXBvciBpbmNpZG1kdW50IHV0IGxhYm9yZSBldCBkb2xvcm
UgbWFn
bmEgYWxpZXVhLiBvdCB1bmltIGFkIGlpbmltIHZlbmlhbSwgcXVpcyBub3N0cnVkIGV4ZX
JjaXRh
dGlvbiB1bGxhbWNVIGxhYm9yaXMgbmlzaSBldCBhbGlxdWlwIGV4IGVhIGNvbWlwZG8gY2
9uc2Vx
dWF0LiBEWlziGFldGUgaXJ1cmUgZG9sb3IgaW4gcmVwcmVoZW5kZXJpdCBpb1B2b2xlcH
RhdGUg
dmVsaXQgZXNzZSBjaWxsZW0gZG9sb3JlIGVlIGZ1Z2lhdCBudWxsYSBwYXJpYXR1ci4gRX
hjZXB0
ZXVvIHNPbnQgb2NjYWVjYXQgY3VwaWRhdGF0IG5vbiBwcm9pZGVudCwgc3VudCBpb1BjdW
xwYSBx
dWkgb2ZmaWNpYSBkZXNlcnVudCBtb2xsaXQgYW5pbnBpZCB1c3QgbGFib3J1bS4=
```

1.4.3

Delivery report

The example below shows a Delivery Status Notification (positive transport confirmation).

Return-Path: <>
Received: from kslv023.esn.kpn.com (10.30.2.23) by kslv019.esn.kpn.com (8.5.113) id 4A7BE27F00000012 for ton@bedrijven.test.otp.nl; Tue, 25 Aug 2009 18:19:56 +0200
Received: by kslv023.esn.kpn.com (8.5.113) id 4A8576E70000001C for ton@bedrijven.test.otp.nl; Tue, 25 Aug 2009 18:19:56 +0200
X-Mailer: OtpTransfer v1.084
From: <postmaster@mail.test.otp.nl>
To: <ton@bedrijven.test.otp.nl>
Message-ID: <M23767e418b9e0b474e6e00f314c10cd3@mail.test.otp.nl>
Date: Tue, 25 Aug 2009 18:19:54 +0200
Subject: Delivery Status Notification (success) for tonox0005@mail.test.otp.nl
MIME-Version: 1.0
Content-Type: multipart/report; report-type=delivery-status; boundary=B6169063283556c1ed30730512e863d00

This multi-part MIME message contains a Delivery Status Notification. If you can see this text, your mail client may not be able to understand MIME formatted messages or DSNs (see RFC 2045 through 2049 for general MIME information and RFC 1891 through 1894 for DSN specific information).

--B6169063283556c1ed30730512e863d00
Content-Type: text/plain; charset=US-ASCII

Your message (id Envy) was successfully delivered to <tonox0005@mail.test.otp.nl>.

--B6169063283556c1ed30730512e863d00
Content-Type: message/delivery-status

Reporting-MTA: dns;mail.test.otp.nl
Original-Envelope-ID: Envy

Final-Recipient: rfc822;tonox0005@mail.test.otp.nl
Action: delivered
Status: 2.0.0

--B6169063283556c1ed30730512e863d00
Content-Type: Text/RFC822-headers

Received: from kslv027.esn.kpn.com (10.30.2.27) by kslv023.esn.kpn.com (8.5.113) id 4A8576E70000001B for tonox0005@mail.test.otp.nl; Tue, 25 Aug 2009 18:19:52 +0200
Received: from localhost.localdomain (10.30.2.16) by kslv027.esn.kpn.com (8.5.113) (authenticated as ton@bedrijven.test.otp.nl) id 4A939F2200000006 for tonox0005@mail.test.otp.nl; Tue, 25 Aug 2009 18:19:52 +0200
From: "Neil Armstrong" <ton@bedrijven.test.otp.nl>
To: "Target" <tonox0005@mail.test.otp.nl>
Message-ID: <fool12@the.dude>
Date: Tue, 12 Mar 18:30:06 2002 +0200
Subject: Zo simpel
MIME-Version: 1.0
Content-Type: text/plain
Content-Transfer-Encoding: base64

--B6169063283556c1ed30730512e863d00--

2 General arrangements

2.1 Standards

2.1.1 Normative RFCs for electronic messaging

Standard	Reference
Simple Mail Transfer Protocol	ftp://ftp.rfc-editor.org/in-notes/rfc2821.txt
Mail Routing and the domain system	ftp://ftp.rfc-editor.org/in-notes/rfc974.txt
SMTP Service Extension for Delivery Status Notification	ftp://ftp.rfc-editor.org/in-notes/rfc1891.txt
SMTP Service Extension for Remote Message Queue Starting	ftp://ftp.rfc-editor.org/in-notes/rfc1985.txt

2.1.2 Normative RFCs for message transport

Standard	Reference
Internet Message Format	ftp://ftp.rfc-editor.org/in-notes/rfc2822.txt
MIME part 1 – Format of Internet Message Bodies	ftp://ftp.rfc-editor.org/in-notes/rfc2045.txt
MIME part 2 – Media Types	ftp://ftp.rfc-editor.org/in-notes/rfc2046.txt
MIME part 3 – Message Header Extensions for non-ASCII text	ftp://ftp.rfc-editor.org/in-notes/rfc2047.txt
MIME part 4 – Registration Procedures	ftp://ftp.rfc-editor.org/in-notes/rfc2048.txt
MIME part 5 – Conformance Criteria and Examples	ftp://ftp.rfc-editor.org/in-notes/rfc2049.txt
MIXER (MIME Internet X.400 Enhanced Relay): Mapping between X.400 and RFC 822/MIME	ftp://ftp.rfc-editor.org/in-notes/rfc2156.txt
Mapping between X.400 and RFC 822/MIME Message Bodies	ftp://ftp.rfc-editor.org/in-notes/rfc2157.txt
Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field	ftp://ftp.rfc-editor.org/in-notes/rfc2183.txt

2.2 Preconditions

These are described in the protocol standards.

2.3 Error messages

These are described in the protocol standards as far as the interaction between MTAs is concerned.

Logical errors that result in rejection of messages and to negative Delivery Status Notifications (DSNs) being sent include:

- Cannot parse the known SMTP headers

- Content-Type not supported
- No body parts available
- More than 1 body part available
- Invalid Base64, quoted printable
- High ASCII in a 7-bit message
- Unknown content-transfer encoding
- Multipart MIME parts cannot be parsed
- Mail addresses (in whichever translated header) are unknown to Digipoort
- Unauthorised linkage (for example: businesses to businesses).
- Logical address is not yet activated.

2.4 Addresses

These are supplied after an account is applied for.

2.5 Limits

These are supplied after an account is applied for.

2.6 Support

Support during connection and use is provided by the Logius Service Centre. See the publisher's imprint for contact details.