



Logius  
Ministerie van Binnenlandse Zaken en  
Koninkrijksrelaties

## Interface description Digipoort File Exchange - FTP

Interface version: 1.6.1  
Document version: 04-10-2018

Date 04 October 2018  
Status Definitive

## Publisher's imprint

Project name      Digipoort

Organisation      Logius  
P.O. Box 96810  
2509 JE Den Haag  
[servicecentrum@logius.nl](mailto:servicecentrum@logius.nl)

Appendix/appendices Schema metafile (metafile.xsd)

## Change history

Date	Author	Interface version	Document version	Description
04-10-2018	Logius	1.6.1	04-10-2018	- Textual modifications made in relation to the exclusivity of the TLS1.2 layer.
13-01-2015	Logius	1.6.1	13-01-2015	- Textual modifications made in relation to the TCP port.
22-12-2015	Logius	1.6.1	22-12-2015	- Textual modifications made and references to the documents modified
26-2-2015	Curtly Inesia	1.6.1	26-2-2015	- Addition of reference to Client FTPs with cURL document - Logius template added
6-6-2012	Tom Breuker	1.6.1	6-6-2012	Appendix with connection details attached.
26-3-2012	Tom Breuker	1.6.1	26-3-2012	- Change history added. - Clarification of naming database and metafile. - Explanation on the ports to be used for data transfer. - Difference between Interface version and Document version is added. Not every new document version should be seen as a modification of the interface. This is, of course, the case the other way around.
8-8-2011	Logius	1.6.1	8-8-2011	Basic version

## Content

<i>Publisher's imprint</i> .....	2
<b>Change history</b> .....	3
<b>Content</b> .....	4
<b>Introduction</b> .....	5
<b>1 Interaction through the interface</b> .....	7
1.1 <i>Transport</i> .....	7
1.2 <i>Contents</i> .....	7
1.2.1 Passive mode (PASV or EPSV) .....	7
1.2.2 Data type .....	7
1.2.3 Organisation per user .....	8
1.2.4 Database.....	8
1.2.5 Metafile.....	8
1.2.6 Delivery of files.....	8
1.2.7 Recommencing interrupted uploads.....	9
1.2.8 Confirmation of receipt.....	9
1.2.9 Error message.....	10
1.2.10 Retrieving files .....	10
1.3 <i>Security</i> .....	11
1.3.1 Transport security .....	11
1.3.2 Confidentiality .....	11
1.3.3 Authentication and authorisation of the client .....	11
1.3.4 Recognised risks and measures .....	12
1.4 <i>Examples</i> .....	12
1.4.1 Setting up a TLS connection .....	12
1.4.2 Locations of files .....	13
<b>2 General arrangements</b> .....	14
2.1 <i>Standards</i> .....	14
2.2 <i>Preconditions</i> .....	14
2.3 <i>Error messages</i> .....	14
2.4 <i>Addresses and parameters</i> .....	14
2.5 <i>Limits and restrictions</i> .....	14

## 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 and business community have to be 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 interface for messages through the File Transfer Protocol (FTP) protocol.

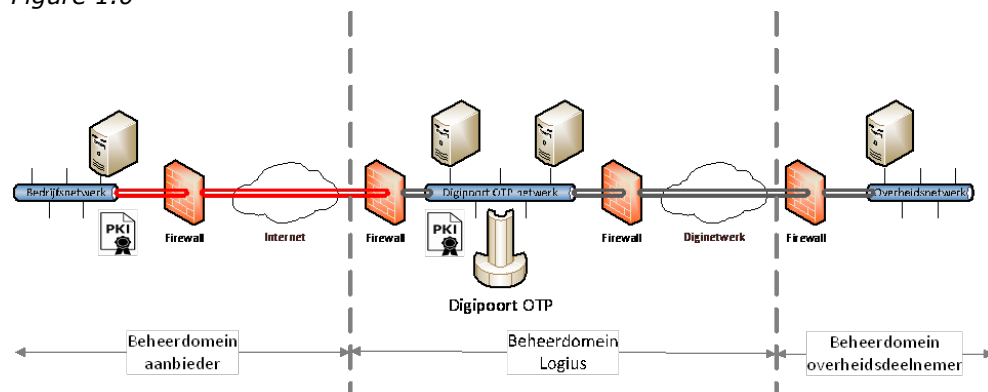
More specifically, this document concerns the 'File exchange FTP – 1.6.1' interface. This interface is intended for the exchange of files between the business community and the government.

Logius also offers the 'Large Messages FTP' interface. This is a separate interface which is not described in this document.

This document is primarily directed at developers of system-to-system linkages between businesses and Digipoort. The provider is responsible for creating the connection through to Digipoort's firewall, the provider's administrative domain as shown in Figure 1.0. It is expected that you have knowledge about FTPs. By way of support, Logius supplies the following documents:

- Date interchange specifications FTPS v1.6.1 Metafile
- Metafile XSD and sample messages
- Interface description FTP\_v1.6.1
- Process description FTPS V.1.6.1
- Connection form technical data - FTPS v1.6.1

Figure 1.0



### **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**

The FTP interface originated from a need for connecting banks and insurers, which have to deliver files to, in the first instance, the Tax and Customs Administration, which were so large that other existing Digipoort interfaces were inadequate.

A joint pilot project involving banks and the Tax Authorities resulted in version 1.0 of these interface specifications. This version was expanded further still to create version 1.6 of the interface specifications. All solutions use the FTP protocol with transport layer security by means of TLS (also known as FTPs).

This interface enables messages to be exchanged between businesses and public authorities. The mutual exchange of messages between public authorities or businesses is not supported.

We expect that the open standards that are used, will develop further in the forthcoming years and that the need for communication will also be subject to change. As a consequence of this new releases of Digipoort will started to be used during the forthcoming years. That can have an impact on the interfaces.

# 1 Interaction through the interface

## 1.1 Transport

The interface can be accessed through a TCP/IP connection. In all cases, for interaction with the interface, a secure connection is created ad hoc. Once the transactions across the interface have been completed, the connection will be disconnected again. The interface allows the delivery of several files in a single session.

Two connections are required for FTP. A test connection which is used to exchange commands and the responses to those commands and a data connection, which is used to exchange the data.

The port (sequences) to be used for the test and data connections are included in the *Connection form technical data - FTPS v1.6.1*

## 1.2 Contents

The principle of the FTP protocol is described in "File Transfer Protocol" – Request for Comments (RFC) 959.

Messages that are sent consist of the file which is sent to the customer (the database) and a file containing metadata concerning the file that is sent (the metafile). The metadata are required to enable the databases to be routed as a message, and to be able to safeguard the integrity and the authenticity of the message. In the other interface supported by Digipoort OTP (SMTP), these metadata form part of the message.

### 1.2.1 Passive mode (PASV or EPSV)

A passive mode is used for the FTP interface. The server does not instigate the establishment of a data connection, but tells the client on which port a connection can be opened to send files. This is done to keep the administrative burden on the users' firewalls as low as possible. The client enforces this using the PASV command or the EPSV command. As a response, the server gives the IP address to be used and the corresponding data port in a series of 6 numbers separated by commas.

Example: 192,168,2,1,34,12

The first four numbers each show one byte of the IP address. This is then 192.168.2.1

The last two numbers in the series together form the port number. The first number multiplied by 256 and the second number is added to that.

This therefore results in  $34 * 256 + 12 = 8716$ .

This is done to ensure that the port number fits in an 8- bits byte.

### 1.2.2 Data type

FTP starts as standard with datatype 'ascii'. Before an upload (STOR) or download (RETR) is started, datatype 'image' (binary) must(!) be switched to using the FTP command TYPE I. This prevents checksums being conducted on various byte series (for example, because different operating systems deal differently with end-of-line characters, etc.).

### 1.2.3 Organisation per user

Every user <sup>1</sup> of the FTP interface receives their own user directory. No other users have access to this directory.

The user directory has two sub-directories:

- in – files and metafiles can be placed here which have to be forwarded by Digipoort to a different user.
- out – files and metafiles can be retrieved here which are delivered by Digipoort on behalf of the user. Digipoort also places receipt confirmations and error messages in this directory.

In each directory, users have restricted rights: requesting a list of files (all directories), placing files (in), retrieving and deleting files (out<sup>2</sup> and in).

### 1.2.4 Database

The content of the database is in accordance with the agreement made between the company and government party.

The following restrictions apply to file names:

- Extensions .meta and .error are reserved. A database supplied by a user may never have one of these extensions.
- File names must fulfil a number of criteria<sup>3</sup>:
  - o they only include characters a-z,A-Z,0-9,\_,., and -
  - o they start with one of the characters a-z,A-Z,\_,.
  - o they have a length of at least 1 and no more than 100 characters, including extensions

### 1.2.5 Metafile

The metafile appears in both incoming and outgoing messages, but depending on the direction can have different content. For the specifications regarding this file, see the attached *Data Interchange Specifications FTPS v1.6.1 Metafile*

All elements occur no more than once.

The definition of the metafile is defined in the XML schema metafile.xsd, appended to these interface specifications.

The same restrictions apply to the file names that apply to the database, with the following addition: A metafile delivered by a user must always have the extension .meta<sup>4</sup>, preceded by the filename of the database, including the extension.

For examples, see the attached sample messages.

### 1.2.6 Delivery of files

First of all, the database is delivered in full (with the STOR command), and then the metafile. Only when the metafile has been delivered in full (recognisable from the end tag) and the database fulfils the metadata provided in the metafile (digest, data reference ID and size), will processing of the message commence and it will be delivered to the addressee (recipient). The database will remain in

<sup>1</sup> NB: In the context of these interface specifications, user means both a business and a government party.

<sup>2</sup> In practice, only an incorrectly placed database can be deleted. When a metafile is placed, this will almost immediately lead to processing the database and metafile.

<sup>3</sup> This corresponds with the following expression as extended regular expression:

/^[a-zA-Z\_][a-zA-Z0-9\_.-]{0,99}(?!\.meta\.error)\z/

<sup>4</sup> This corresponds with the following expression as extended regular expression:

/^[a-zA-Z\_][a-zA-Z0-9\_.-]{0,99}(?!\.meta)\z/



the user directory until the corresponding metafile is placed and found to be correct. At that time, both will be deleted from the sender's user directory. This is confirmed with a receipt confirmation (see 1.2.8). If delivery is unsuccessful, an error message will be given (see 1.2.9).

**1.2.7**      *Recommencing interrupted uploads*

The FTP interface allows restoration of an upload has been interrupted by using the REST (restart) command.

**1.2.8**      *Confirmation of receipt*

As soon as it has been established that the files that were delivered fulfil the interface specifications, a receipt confirmation will be placed in the sender's "out" directory.

The receipt confirmation consists of a copy of the metafile, plus the element 'received' with the time stamp showing the time at which the message is delivered.

The name of the receipt confirmation is composed as follows:

{data-reference id verzender}\_{data-reference id Digipoort}.ok

Example:

RENSAGEG\_20081231\_Ambobank\_Amstelhoven.xml\_Digipoort\_018b0112-8171-4fed-b360-aa509ba1c6a9.ok

If this file name is longer than the maximum permitted length, the "data-reference\_id sender" is cut off.

#### 1.2.9

##### *Error message*

When Digipoort is unable to process the metafile and database combination that is offered, an error message is placed in the sender's "in" directory<sup>5</sup>.

The name of the file in which the error message is included is built up as follows:

```
{data-reference id verzender}_{data-reference id Digipoort}.error
```

Example:

```
RENSAGEG_20081231_Ambobank_Amstelhoven.xml_Digipoort_018b0112-8171-4fed-  
b360-aa509ba1c6a9.error
```

If this file name is longer than the maximum permitted length, the "data-reference\_id sender" is cut off.

#### 1.2.10

##### *Retrieving files*

Digipoort first delivers the database, then the metafile. Retrieval of files (using the RETR command) can start as soon as the metafile is visible. After they have been downloaded successfully, the files can be deleted by the user (using the DELE command)<sup>6</sup>.

---

<sup>5</sup> The processing of the message can be restarted by placing a corrected metafile, without the database having to be re-uploaded.

<sup>6</sup> The interface assumes that the user will not use delete until he is satisfied that the retrieval of the file was successful. After delete has been used, the file can no longer be retrieved

## 1.3 Security

### 1.3.1 *Transport security*

To secure the transport, the Transport Layer Security (TLS) is used. Both the server's certificate and the client's certificate are used to set up a symmetrical secure connection. The principle of TLS connections for the FTP protocol are described in "Securing FTP with TLS" - RFC 4217.

To ensure that the TLS connection is successful, a PKIo certificate has to be used, in which case the certificate:

- is valid
- does not appear on a Certificate Revocation List
- is registered with Logius

The data connection must always be secured with the Protect (PROT) command at Private (P) level of security. The server does not allow commands that use the data connection before the PROT command is given and the level is set at P. If the PROT command has not yet been given, the server's response to commands that use the data connection is always an error code.

Using the Clear Command Channel (CCC) command, the test connection can be returned to a plain text status. The server refuses the CCC command because this provides an opening for Man-In-The-Middle (MITM) attacks and always responds to the request with an error code, as specified in RFC 4217.

By using TLS, the Protection Buffer Size (PBSZ) command is still mandatory but a value of '0' always has to be given, which shows that this relates to a streaming connection.

### 1.3.2 *Confidentiality*

Confidentiality is achieved by the restrictions that are placed on the user and the various directories. One user is given access to just one directory. Government organisations are not given access to the director of a business. Messages are only delivered to the government to which they are addressed.

### 1.3.3 *Authentication and authorisation of the client*

The client has to authenticate itself by means of a username and a password prior to authorisation being given for access to the client's own directory. When the TLS connection is made, the server checks the username against the client certificate that is used (see 3.3.1). The access is denied if the username does not correspond with the certificate.

#### 1.3.4 *Recognised risks and measures*

The table shown below provides an overview of generally identified risks in relation to the use of the FTP protocol and the measures that have been taken to mitigate these risks.

Risk	Measure
All commands that are given by the client before the AUTH TLS command is given are in plain text and are at the expense and under the responsibility of the user. Authorisation attempts on an insecure connection mean that usernames and passwords can be read by third parties.	If a sufficiently secure connection has not yet been set up, at the very most, a client may give the commands HELP, FEAT and AUTH.
The use of the standard FTP user (anonymous) gives opportunities for unauthorised clients to retrieve files that are intended for other parties.	The use of the anonymous user is not permitted.
To place these on the FTP server of GBO and to retrieve these at a later stage.	
The use of the proposed digest gives no guarantees for the integrity of the file, it could be compromised without this being evident to the recipient.	Digipoort complies with a large number of information security measures which limit the possibility of files being compromised to a minimum. These security measures are verified annually during an independent audit. If full certainty about the integrity (and authenticity) is required, the sender and recipient can agree to apply a signed digest or encryption to the file.

### 1.4 **Examples**

NB the IP addresses and port numbers that are used can differ in actual practice.

#### 1.4.1 *Setting up a TLS connection*

```
<server is waiting for a tcp connection on port 21>
<client opens a connection on tcp/21>
S: 220 oe.procesinfrastructuur.nl ready
C: AUTH TLS
S & C: <TLS connection is being negotiated and set up>
S: 234 Security data exchange complete
```

```
C: PBSZ 0
S: 200 Protection Buffer Size set to streaming
C: PROT P
S: 200 Data connection is private
C: USER bank
S: 331 Username      OK, need password for login
C: PASS "the bank's password"
S: 230 User logged in, proceed.
C: QUIT
<server closes the connection>
```

#### 1.4.2 *Locations of files*

```
<the server and client have made a TLS connection, client is logged in>
S: 230 User logged in, proceed.
C: EPSV
S: 227 Entering passive mode (144,43,253,65,82,8)
C: TYPE I
S: 200
C: STOR RENSAGEG_20081231_Ambobank_Amstelhoven.xml
C: <Opens a TCP connection at 144.43.253.65 port 21000>
S: 150 Opening BINARY mode SSL data connection for

RENSAGEG_20081231_Ambobank_Amstelhoven.xml.
C: <TLS handshake and encrypted transmission of the data.>
S: 226 Transfer complete.
C: <repeats the steps from STOR for files to be
placed>
C: QUIT
<server closes the connection>
```

## 2 General arrangements

### 2.1 Standards

Standard	Reference
TCP & TCP/IP	<a href="http://www.rfcsearch.org/rfcview/RFC/675.html">http://www.rfcsearch.org/rfcview/RFC/675.html</a> <a href="http://www.rfcsearch.org/rfcview/RFC/1958.html">http://www.rfcsearch.org/rfcview/RFC/1958.html</a> <a href="http://www.rfcsearch.org/rfcview/RFC/1122.html">http://www.rfcsearch.org/rfcview/RFC/1122.html</a>
File Transfer Protocol	<a href="http://www.rfcsearch.org/rfcview/RFC/959.html">http://www.rfcsearch.org/rfcview/RFC/959.html</a>
Transport Layer Security v1.2 (TLS)	<a href="http://www.rfcsearch.org/rfcview/RFC/5246.html">http://www.rfcsearch.org/rfcview/RFC/5246.html</a>
Securing FTP with TLS	<a href="http://www.rfcsearch.org/rfcview/RFC/4217.html">http://www.rfcsearch.org/rfcview/RFC/4217.html</a>

### 2.2 Preconditions

The client certificates to be used to set up the secure connection are PKI.Overheid certificates for use by services (see <http://www.pkioverheid.nl/voor-certificaatverleners/programma-van-eisen/deel-3b>).

The user is responsible for the purchase of a client certificate from a service provider appointed by PKI.Overheid.

### 2.3 Error messages

All applicable error messages for setting up the FTP connection and the exchange of files using FTP, are described in the aforementioned standards.

Logical errors that result in the rejection of files are, amongst others:

- A missing database for which a metafile is present
- The metafile cannot be read, or has an incorrect structure
- Missing mandatory elements in the metafile
- Sender unknown in the incoming metafile
- Receiver unknown in the incoming metafile
- File size of the database does not correspond with the file size in the metafile
- Digest of the datafile does not correspond with digest value in the metafile
- Filename of the database or metafile is invalid (does not fulfil the file naming conventions)

### 2.4 Addresses and parameters

These are supplied after an account is applied for.

### 2.5 Limits and restrictions

Technical restrictions of the interface are supplied after an account is applied for.