

Title:  
**TRS Requirement  
specification**

doc type

**Requirement specification 3.0**

Created:  
**8/29/2008 11:12:00 AM**

Last changed on:  
**3/16/2011 3:00:00 PM**

Author:  
**TRS**

Last changed by:  
**TRS**



---

# TRS Requirement Specification



## Revision history

Version	Date	Remarks
0.1	2006	Initial document
1.21	2006-11-02	First English version
1.33	2007-01-15	Translation of Swedish version 1.33.
1.34	2007-01-18	Minor changes after review by Tobias. Footnotes added. Certificate changed to key in 2.5 ExternalFileName fixed in 2.4 Diagram in 2.2.2 translated
1.35	2007-03-26	Changes in explanation of filenames and Feedback fileextensions Bank day not validated in filenames
1.36	2007-04-11	Feedback filenames changed
1.37	2007-04-17	Feedback files will be encrypted "Status file" changed to "Feedback file"
1.38	2007-04-25	Corrections in filename conventions
1.39	2007-05-28	Venue specific information added PKI information updated Other minor corrections
1.40	2007-05-28	Minor corrections
1.41	2007-06-07	Detailed description of feedback file contents.
1.42	2007-06-11	Changes made to the explanation of the XML format under 2.4.2.
1.43	2007-06-25	References to report format documents changed in 2.4. Clarification on several RP transactions in one file from TRP in 2.4
1.44	2007-08-23	Statuscodes updated Encrypted feedback files updated Xml namespace – only one PKI workflow updated "Update transaction" type removed
1.45	2007-10-5	Minor typo corrections in: 1.3.3, 1.4, 1.4.1, 1.4.5, 1.4.9, 1.4.14.2, 1.5.1, 1.5.1, 1.6.2, 1.10.3, 1.10.13.2, 1.10.13.2; 1.11.1, 1.11.13.2, 1.11.13.2.  1.4.10 status file removal 1.4.12, 1.19.10, 1.11.11 SEQ clarified 1.4.14, 1.11.13 RTAC added 1.10.13.1 ICA added 1.10.3 TRFID clarified and example changed

Title:  
**TRS Requirement  
specification**  
doc type  
**Requirement specification 3.0**

Created:  
**8/29/2008 11:12:00 AM**  
Last changed on:  
**3/16/2011 3:00:00 PM**

Author:  
**TRS**  
Last changed by:  
**TRS**



---

2.0	2008-09-01	Major changes to the structure. Concentrated general information about the system first and then sections describing each type of possible report to send to the TRS system.  Added section about reporting Customer Client information.
2.1	2009-02-13	Added section 1.10.1.2 File Content
3.0	2011-03-11	Added OTC transactions. Sections that describes information specific to Sweden marked with (Sweden only).



## Table of content

1	Problem-oriented system description .....	6
1.1	Overview of the TRS solution.....	6
1.1.1	Reporting Firm, RF .....	7
1.1.2	Technical Reporting Firm, TRF .....	7
1.1.3	Trading Venue.....	7
1.1.4	Database for Collection (TRS IN).....	7
1.1.5	TRS Data Storage.....	7
1.2	Reporting.....	8
1.2.1	Reporting.....	9
1.2.2	Feedback.....	10
1.3	Transfer Model and Transport of Data.....	11
1.3.1	Transfer via HTTP(s) (HTTP(s) upload) .....	11
1.3.2	Transfer via FTP.....	11
1.3.3	Manual input – Web Input .....	12
1.4	FI PKI Certificate based file security .....	13
1.4.1	Certificates .....	13
1.4.2	File structure.....	14
1.5	FI PKI Workflows for Reporting Party.....	15
1.5.1	Encrypt and submit a transaction file.....	15
1.5.2	Retrieve and decrypt a status file .....	15
1.6	FI PKI non functional requirements.....	17
1.6.1	Encryption .....	17
1.6.2	Certificate .....	17
1.6.3	Checksum and signing .....	17
1.6.4	Compression.....	17
1.6.5	Compatibility.....	17
1.6.6	Questions and answers .....	17
1.7	Transaction report file (Mifid transactions) .....	19
1.7.1	File format (XML) .....	19
1.7.2	Filename .....	19
1.7.3	File type.....	20
1.7.4	File size .....	21
1.7.5	Feedback file .....	21
1.7.6	Client Information .....	23
1.8	RF administration interface.....	24
1.8.1	Log in .....	24
1.8.2	History and Statistics.....	24
1.8.3	Account information .....	24
1.9	Venue Transaction report file (venue reporting only).....	25
1.9.1	File format (XML) .....	25
1.9.2	Filename .....	25
1.9.3	File type.....	26
1.9.4	Feedback file .....	27
1.10	Instrument file (venue reporting only) .....	29
1.10.1	File format (XML) .....	29



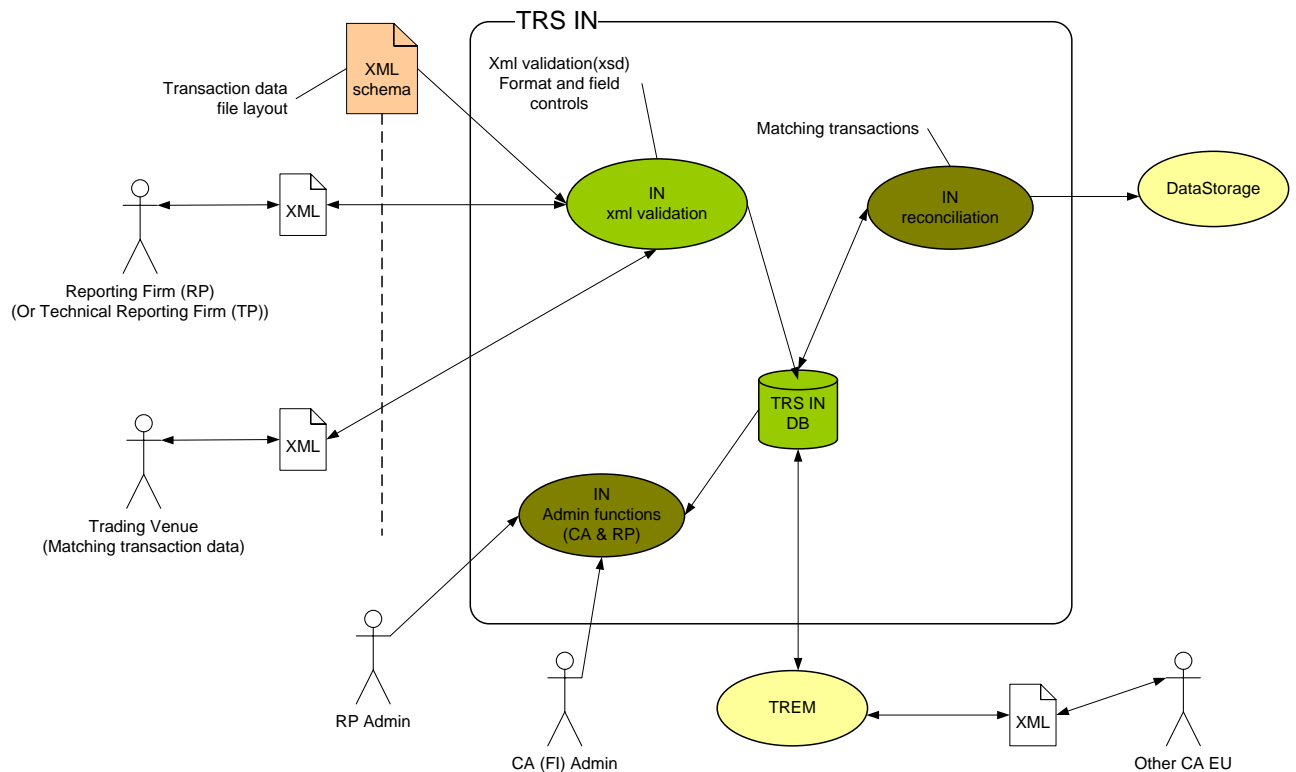
---

1.10.2	Filename .....	29
1.10.3	File type.....	30
1.10.4	Feedback file .....	31
1.11	Customer Client Information reports .....	33
1.11.1	File format (XML) .....	33
1.11.2	Filename .....	33
1.11.3	File type.....	34
1.11.4	Feedback file .....	35
1.12	Transaction report file (OTC transactions) .....	38
1.12.1	File format (XML) .....	38
1.12.2	Filename .....	38
1.12.3	File type.....	39
1.12.4	Feedback file .....	40
1.12.5	Client Information .....	42

# 1 Problem-oriented system description

## 1.1 Overview of the TRS solution

The TRS - Transaction Reporting System is described below in a schematic outline with the primary players:



*(TRS reporting model overview)*



### **1.1.1 Reporting Firm, RF**

The reporting investment firm with reporting obligation shall submit reports to the system daily or on request. The reporting concerns all purchase and sales transactions for instruments or underlying instruments that have been admitted to trading on a regulated market. Transactions (Mifid) shall be reported daily, if a reporting firm has no transactions to report an empty file should be submitted. OTC Transactions shall be reported daily, if a reporting firm has no OTC transactions to report an empty file could be submitted but that is no obligation. Occasionally Customer Client information will be requested.

### **1.1.2 Technical Reporting Firm, TRF**

Reporting firms (RF) may allow another firm to do the technical reporting. Such a “Technical Reporting Firm” (TRF) of this kind could be a regulated market or MTF, or other player that the competent authority authorises to report on behalf of other firms. If the RF is doing their reporting themselves they assume the role as their own TRF.

### **1.1.3 Trading Venue**

The regulated markets also play a role as supplier of “answer key” data for transactions that have been executed. This data will be used for crosschecking the reported transactions. They shall also submit information regarding the instruments that they have admitted to trading.

### **1.1.4 Database for Collection (TRS IN)**

All data is collected in the TRS IN database. Received data is validated to ensure that it is correct. Several controls are exercised, including format validation and data validation, which take place immediately on the same occasion.

### **1.1.5 TRS Data Storage**

When data has been quality assured in TRS IN, it is passed on and stored in the “TRS Data Storage database”.

The data will be used for identifying signals and trends that calls for additional analyses and investigations. As support in investigations and ad hoc analyses, an analytical system will also be connected to the database.

Title:  
**TRS Requirement  
specification**  
doc type  
**Requirement specification 3.0**

Created:  
**8/29/2008 11:12:00 AM**  
Last changed on:  
**3/16/2011 3:00:00 PM**

Author:  
**TRS**  
Last changed by:  
**TRS**



---

## 1.2 Reporting

Reporting is done on a daily basis by the reporting institutes or on request. There are three services / interface that make this possible. When TRS has received the report a detailed feedback file will be returned as a confirmation of the submitted report. A report must be validated against the current xml schema corresponding to the report before sending it to the TRS system.

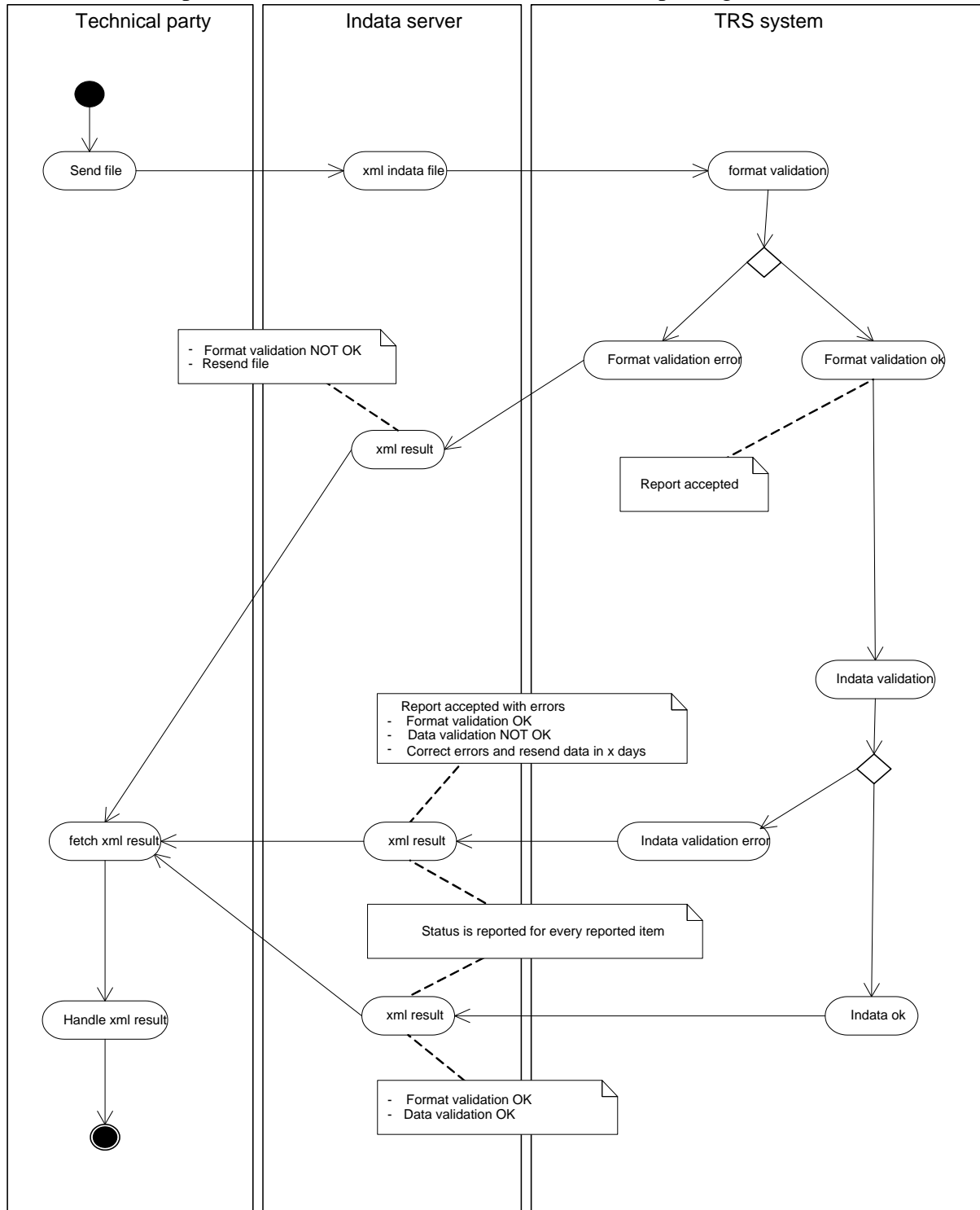
The services/interfaces are:

- Upload via FTP
- Upload via HTTP(s)
- Manual data entry via HTTP(s)



## 1.2.1 Reporting

Below is a description of the inflow of data into TRS from a reporting firm:



(TRS Activities)



#### Main flow

1. All received reports are validated for format
2. Status for format validation becomes “OK” (alt flow:2)
3. The report is approved and the task of reporting is finished.
4. Data validation at record level
5. Status for all records becomes “OK” (alt flow:5)
6. Status for the report becomes “file accepted”
7. The flow is completed

#### Alternative flow

2. Format validation NOT OK
  - a. Status for the report becomes “rejected”.
  - b. New and corrected report is expected.
  
1. Data validation of transaction NOT OK
  - a. Each record receives status OK/error with associated error code in the feedback report.
  - b. Status for the report becomes “accepted with errors”.
  - c. Prepared records are expected to be sent to FI after 3 days.

### 1.2.2 Feedback

When a reporting firm has sent a report to the TRS system via FTP, a feedback file is generated and placed in the feedback folder on the FTP server. To minimize network load the file is zipped and the zip file will have the same name as the feedback file itself but with the extension .zip.

When a reporting firm has sent a report to the TRS system via HTTP a summary of the feedback is shown to the user on the webpage. There will also be a hyperlink to a download page where the logged in user can download the feedback file.

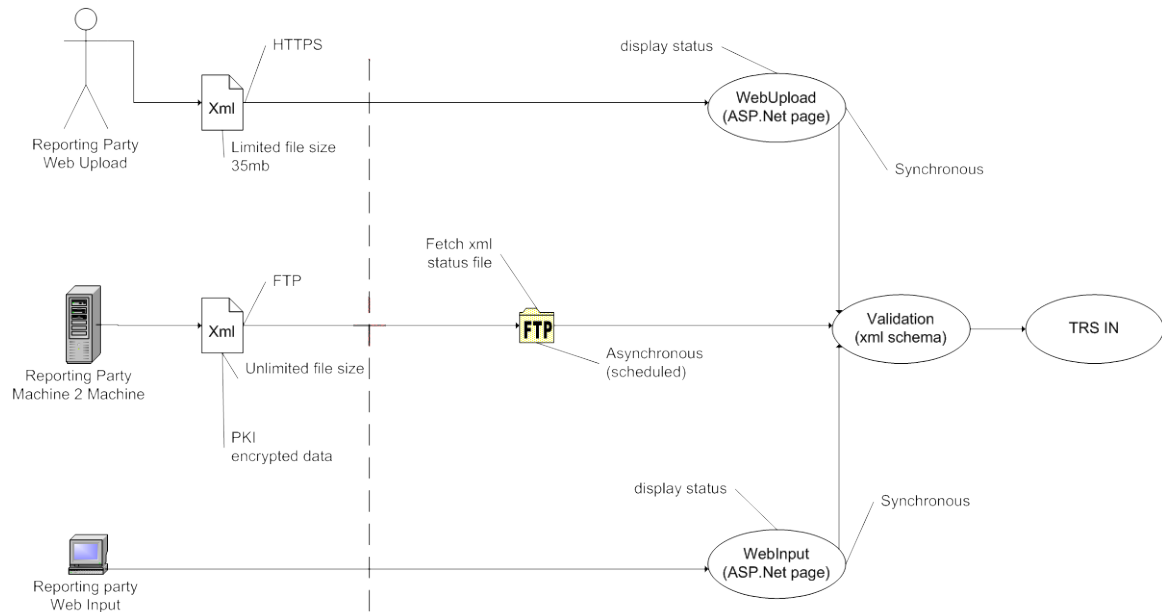
The actual feedback file is of type XML and will follow the schema located at <http://schemas.fi.se/trs/>. If an unexpected error has occurred the feedback file will be based on the schema located at <http://schemas.fi.se/trs/OutGeneric/>.

The purpose of feedback file is to inform the reporting party about the status of an uploaded report. Feedback is always returned when a report is sent to the TRS system. The competent authority expects all reporting firms to take necessary actions when feedback is returned. Feedback files are by default unencrypted. Please contact your competent authority for ordering encrypted feedback files. Encrypted feedback files can only be generated if the incoming file is encrypted and the file has the status “ACC” (Accepted) or “AWE” (AcceptedWithErrors).

### 1.3 Transfer Model and Transport of Data

Data may be transferred in three different ways; the reporting firm may choose any of these alternatives. Below is a description of these alternatives.

TRS data transfer method PKI



(TRS data transfer method PKI)

#### 1.3.1 Transfer via HTTP(s) (HTTP(s) upload)

(Sweden only)

Finansinspektionens existing infrastructure for e-ID will be used for the identification process. Web upload has a maximum limit for the size of data files, currently estimated at 35 megabytes.

Using certificates makes it simple to communicate between browser and server via an encrypted connection (128-bit SSL), which may be sufficient for privacy-protected transfer so files would not have to be encrypted before transfer. However, it may be of interest to compress data prior to transferring data, in which case files can be zipped. Files can also be divided to reduce file size and increase transmission speed.

#### 1.3.2 Transfer via FTP

Upload of reports via the FTP service requires access to the FTP server.

After the login is accepted each TRF will be placed in their own "home-folder". Two subfolders "Upload" and "Status" will be present. TRF stores the reported file in the "Upload" folder. Shortly afterwards (depending on file size) a feedback file will be stored in

Title:  
**TRS Requirement  
specification**  
doc type  
**Requirement specification 3.0**

Created:  
**8/29/2008 11:12:00 AM**  
Last changed on:  
**3/16/2011 3:00:00 PM**

Author:  
**TRS**  
Last changed by:  
**TRS**



---

the “Status” folder by the system. At the same time the uploaded file will automatically be removed by the system. As a security solution for automatic file transfer a PKI and certificate based solution will be used. With the digital certificate the data is signed, encrypted and compressed before it is sent to TRS. Data is securely sent via open FTP. This provides a simple and standardized possibility for automation. It also secures identity and integrity. See 2.5 requirements and handling of files with certificate.

### **1.3.3 Manual input – Web Input**

It is possible to manually enter Mifid transaction data and client information directly on a website, though this is only appropriate for an extremely limited number of records. Alternatively, the reporting firm must manually generate report files in accordance with the defined formats. See chapters 1.7 and 1.11 .



## 1.4 FI PKI Certificate based file security

(Sweden only)

The use of certificates will make it possible to sign, encrypt and identify transaction and feedback files that are transferred to and from Finansinspektionen. Finansinspektionen has chosen open standards as much as possible and with that facilitate for the reporting party to integrate the solution in their system environment. The implementation has been verified to work with an X509 version 3 Certificate structure created in house with the OpenSSL suite, but also with a managed X509 version 3 Certificate structure supplied by Telia.

Below you will find the requirements and work steps that are required in order to use certificate based file security together with Finansinspektionen.

### 1.4.1 Certificates

Finansinspektionen has chosen to use X509 version 3 standard certificates that are based on the RSA and SHA1 algorithms. This is the same type of certificates that are used to secure websites (HTTPS). Finansinspektionen has chosen Telia as the Certificate Authority. Other Certificate Authorities may have identical or similar routine when working with certificates.

To acquire a valid certificate the reporting party needs to perform the following steps:

1. The reporting party visits the Enrolment site. Finansinspektionen will provide a website where the reporting party can request a Certificate. The website will request the reporting party to fill out some specific fields. Finansinspektionen will provide instructions on how to fill out the different fields. During this step a private key and public key will be stored in the reporting party's machine; the certificate request (containing the public key) will be automatically submitted thru the website to Finansinspektionen.
2. Finansinspektionen will then contact the reporting party by a secure way to verify that they are responsible for the certificate request. When the authenticity of the certificate request has been established, the reporting party will receive an email with a one time password and an URL to a page where they can retrieve their signed certificate.
3. The reporting party will visit the supplied URL and install the signed certificate in their machine. This must be performed on the same machine and with the same user account as was used in step 1. Now the reporting party has acquired a valid certificate. Next step is to export it from the machine into a file.
4. The reporting party will use the "Microsoft Management Console" (MMC) and the snap-in module called "Certificate" to export the certificate into a file. The certificate will be stored in "Console Root" -> "Certificates - Current User" -> "Personal" -> "Certificates". Right click on the certificate and choose "All activities" -> "Export". The reporting party will now be forced to answer a few questions on how to export the certificate:
  - Export private key = **Yes**
  - Activate strong protection (demands IE5, NT4) = **No**
  - Use a strong password



5. The certificate (both private and public key) should now be stored on file in PKCS12 format, that Microsoft call Personal Information Exchange (.PFX). Because this file includes the private key it should be stored in a secure way with a good password. Depending on the system environment the reporting party uses they may need to perform additional conversion of the file to suite their particular system. Following here are examples showing some common types of conversion performed with the OpenSSL suite:

- Convert PKCS12 format certificate to PEM format certificate  
*openssl pkcs12 -in Cert-PKCS12.pfx -out Cert-PEM.pem -nodes -clcerts -nokeys*
- Export private key from PKCS12 format to RSA private key  
*openssl pkcs12 -in Cert-PKCS12.pfx -out Private-key.key -nodes -nocerts*

Except for their own certificate the reporting party needs two more certificates. The first is the CA certificate; the reporting party will use that to verify the signature on the FI certificate. The second is the FI certificate; the reporting party will use that to encrypt the symmetric encryption key and initializing vector.

#### 1.4.2 File structure

Finansinspektionen has chosen a simple file structure for the reporting party to use, the goal with the solution is that it would be possible to use the structure in different system environments. The files are named in such a way it should be easy to understand their purpose in the structure. The file structure consists of the following files:

- **Key.enc** (format: binary)  
The file includes an encrypted symmetric encryption key. The key has been encrypted by the user with the FI certificate.
- **Iv.enc** (format: binary)  
The file includes an encrypted initializing vector. The vector has been encrypted by the user with the FI certificate.
- **Data.digest** (format: PKCS1)  
The file includes a signed checksum of the encrypted file. The checksum has been signed by the user.
- **Data.cer** (format: PEM)  
The file includes the user certificate in PEM format. This is necessary so FI can verify the signed checksum without storing all user certificates locally. Finansinspektionen will also check if the certificate has a valid signature by a trusted CA, and if the certificate is revoked or expired.
- **Encrypted data file** (format: binary)  
The file includes the encrypted file the user wants to submit to Finansinspektionen.

There are many advantages with this file structure compared to using a more or less open format where all the different parts are mixed into one file.

- Easy to understand what each file contain and the purpose of it.
- The reporting party doesn't need to invest in a commercial tool for example PGP.
- Easy to deploy in many types of different system environments.



## 1.5 FI PKI Workflows for Reporting Party

(Sweden only)

### 1.5.1 Encrypt and submit a transaction file

1. Compress XML-file <TR\_BIC\_YYYYMMDD\_SEQ.XML> with ZIP <TR\_BIC\_YYYYMMDD\_SEQ.ZIP>.
2. Generate a 256 bit random key.
3. Generate a 128 bit random initializing vector.
4. Encrypt the compressed file from step 1, with the key from step 2 and initializing vector from step 3. The encrypted file should be stored in binary form with the following filename: "<original filename>.encrypted", the encoding of the file must be binary. For example TR\_BIC\_YYYYMMDD\_SEQ.XML.ZIP.encrypted.
5. Encrypt the key from step 2 with FI public certificate. The encrypted key must be stored in binary with the following filename: "key.enc".
6. Encrypt the initializing vector from step 3 with FI public certificate. The encrypted initializing vector must be stored in binary with the following filename: "iv.enc".
7. Create a checksum of the encrypted file from step 4.
8. Sign the checksum from step 7 with your private key and save the result in "data.digest". The format of the file must be PKCS1.
9. Save your own public certificate PEM formatted with the following filename: "data.cer". **Do NOT include the private key!**
10. Compress the following files with ZIP.
  - Encrypted file from step 4
  - Encrypted key, "key.enc"
  - Encrypted initializing vector, "iv.enc"
  - Signed checksum, "data.digest"
  - Your certificate, "data.cer"
11. Rename the file according to the pattern TR\_BIC\_YYYYMMDD\_SEQ.ENC
12. Upload the compressed file from step 10 to the FTP server at Finansinspektionen.

### 1.5.2 Retrieve and decrypt a status file

1. Download the compressed file from the FTP server at Finansinspektionen.
2. Decompress the file.
3. Identify the name of the encrypted status file, the filename will end with ".encrypted"
4. Verify FI certificate stored in "data.cer", it's important to check if the certificate has been revoked och expired. Finansinspektionen will provide a Certificate Revocation List, the distribution point of the list will be listed as a CRL extension in every certificate signed by Finansinspektionen. It lies at the responsibility of the reporting party to perform a valid verification of the certificate that the reporting party can trust. **Finansinspektionen will never use a revoked or expired certificate!**
5. Verify the signed checksum in "data.digest" using the FI certificate stored in "data.cer".

Title:  
**TRS Requirement  
specification**

doc type

**Requirement specification 3.0**

Created:  
**8/29/2008 11:12:00 AM**

Last changed on:  
**3/16/2011 3:00:00 PM**

Author:  
**TRS**

Last changed by:  
**TRS**



- 
6. Decrypt the encrypted symmetric key stored in “key.enc” with your private key.
  7. Decrypt the encrypted initializing vector stored in “iv.enc” with your private key.
  8. Decrypt the encrypted status file using the key from step 6 and initializing vector from step 7.



## **1.6 FI PKI non functional requirements**

(Sweden only)

### **1.6.1 Encryption**

The reporting party must use the AES algorithm in CBC-mode (Cipher Block Chaining) with a 256 bit key and a 128 bit initializing vector. The key and initializing vector must be randomly generated for each reporting occasion. The AES algorithm is specified in FIPS PUB 197<sup>1</sup>.

### **1.6.2 Certificate**

Telia will act as Certificate Authority commissioned by FI. The Certificates will be standard X509 version 3 certificates. The X509 certificate standard uses the RSA algorithm with a 1024 bit key length, and the SHA1 algorithm as hash algorithm. The reporting party will be able to acquire a certificate by following the steps mentioned in chapter 1.1.1. FI will also provide pre generated certificates. All certificates will be delivered in PKCS12-format. There is no certificate request fee for the reporting party.

### **1.6.3 Checksum and signing**

The checksum of the encrypted file must be created with the SHA1 algorithm. The signed checksum must be stored in PKCS1 format. The SHA1 algorithm is specified in FIPS PUB 180-1<sup>2</sup>.

### **1.6.4 Compression**

All compression must be done with ZIP.

### **1.6.5 Compatibility**

Finansinspektionen has verified compatibility in a Windows X86 environment between Microsoft .NET Framework 2.0, OpenSSL version 0.9.8d and Java 1.6.0.1.

Finansinspektionen has also produced example code for the reporting parties with the following environments: Microsoft .NET Framework 2.0, OpenSSL version 0.9.8d and Java 1.6.0.1, available on request.

### **1.6.6 Questions and answers**

- Q: Why can't the reporting party use their certificate and the FI certificate to directly encrypt the XML-file?  
A: X509 Certificates contains a 1024 bits RSA public key. The RSA algorithm works in such way that if we would use it to encrypt the XML file directly it would give an attacker significant advantage when trying to break the encryption. Because of this

---

<sup>1</sup> <http://www.csrc.nist.gov/publications/fips/fips197/fips-197.pdf>

<sup>2</sup> <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf>



---

issue we have decided to use the AES algorithm to encrypt the file and then encrypt the small AES key with RSA. The AES algorithm is another type of algorithm that is more suited to encrypt large files.

- Q: Will the reporting party's certificate or FI's certificate expire some day?  
A: Yes. All certificates have a limited life time. It's important that the reporting party is familiar with the procedures mentioned above on how to acquire a new certificate and plan well ahead.
- Q: Our certificate/private key/system has been compromised! What should we do?  
A: You should call the telephone number give to you by FI and request an immediate revocation of the certificate.



## 1.7 Transaction report file (Mifid transactions)

The content of the transaction report is based on MiFID and must adhere to given requirements regarding:

- File format (Valid XML)
- Filename
- File type
- File size

Transactions from several Reporting Parties can be sent together in one file by a Technical Reporting Party or in one file for each Reporting Party, whichever is preferred.

### 1.7.1 File format (XML)

File format is defined as the XML structure that the transaction report must comply to. It is defined in the XML-schema file “TransactionReport.xsd”. All fields in this structure are described in the document “TRS Report Format for reporting party”.

#### 1.7.1.1 Information type

There are two different information types in a transaction report file:

1. Transaction Record Info - transaction information
2. Cancellation Transaction Type – cancellation of a transaction

Transactions with the same value in the field “TransactionReferenceNumber” must only be present once for each information type (transaction or cancellation) in a single file.

### 1.7.2 Filename

The filename is according to following structure:

TR\_TRFID\_YYYYMMDD\_SEQ.TYPE

Section	Explanation	Remarks
<i>TR</i>	Indicates that the file is a Transaction report	
<i>TRFID</i>	BIC code for Technical Reporting firm	
<i>YYYYMMDD</i>	Reporting date that the transaction file refers to	
<i>SEQ</i>	Sequence number	A numerical 4 digits sequence number [0000-9999]. Must be unique per day.
<i>TYPE</i>	File type	See 1.7.3 for valid file types



**Example:**

Name on transaction report: *TR\_FISSSEBANK1\_20070601\_0001.XML*

*1.7.2.1 Incorrect filename*

If a transaction report with an incorrect filename is reported the process flow of the file will be aborted and an error will be created in a feedback file

**1.7.3 File type**

Depending on how the transaction report is -transferred -uploaded via HTTP or FTP the following file types are accepted.

File type	Interface		Remark
	HTTP	FTP	
XML	✓	✓	Se 1.7.3.1
ZIP	✓	✓	Se 1.7.3.2
ZIP (ENC)	✗	✓	Se 1.7.3.3

✓ = Accepted

*1.7.3.1 XML file*

The uploaded XML file must comply with the specified XML-schema. The content of the transaction report must be in a valid xml-format with Utf-8 encoding and it must be validated against the current xml schema before the file is sent to the TRS system. XML-schema files are available on <http://schemas.fi.se/trs> for download.

Please note that only one namespace is allowed for TRS XML files. This means that the root element must consist of exactly two attributes: namespace and version. The prefix of the root element must correspond to the declared namespace. See example below:

```
<trs:report xmlns:trs="http://schemas.fi.se/TRS/InTrans" Version="3.0">
```

*1.7.3.2 Compressed files*

The uploaded compressed file must comply with following:

- The file type must be ZIP and must only contain ONE compressed file (and no subfolders).



- The compressed file must be a XML file.
- The compressed file must use the same filename as the ZIP-file but with the ZIP extension.

### 1.7.3.3 Signed and encrypted files (ENC)

Signed and encrypted files are created with a certificate. See 1.4 for information about how to sign and encrypt files.

### 1.7.4 File size

Depending on if the transaction file is uploaded via HTTP or FTP there are size restrictions:

File type	Interface	
	HTTP	FTP
XML	35 Mb	No limit
ZIP	200 Kb	No limit
ZIP (ENC)	✘	No limit

If a transaction file is too large when uploaded via http the transaction flow will be aborted and an error will be generated in the feedback file.

### 1.7.5 Feedback file

The feedback file is the XML-file that will be returned to the Technical Reporting Firm after upload, validation and verification of the transaction file.

The returned feedback file will be compressed and stored in the “status” folder. If the transaction file was encrypted the feedback file will also be returned as encrypted.

Note: Encrypted feedback files must have been ordered from your competent authority and the transaction file must have either status “ACC” (Accepted) or “AWE” (AcceptedWithErrors) in order for the TRS system to return an encrypted feedback file.

The feedback files will be removed from the status folder after 10 days.

The transactions in the feedback file are identified by the transaction reference number of the submitted transaction. For cancellation transaction records, that do not have a transaction reference number, this number is the same reference number as the transaction referenced to be cancelled.

#### 1.7.5.1 Name of the feedback file – if the transaction report has a correct filename

The name of the feedback file will be generated by the system and will reflect the name of the uploaded transaction file.



When a transaction report has been uploaded with a correct filename the returned feedback file will have the same name with prefix FB. The returned file will be compressed.

**Example:**

Name on the reported transaction file: *TR\_FISSSEBANK1\_20070601\_0001.XML*  
Name on the associated feedback file: *FB\_TR\_FISSSEBANK1\_20070601\_0001\_0001.XML*  
Name on the compressed feedback file: *FB\_TR\_FISSSEBANK1\_20070601\_0001\_0001.ZIP*

*1.7.5.2 Name of the feedback file – if the transaction file has an incorrect filename*

If the uploaded transaction file has an incorrect filename the transaction flow will be aborted and an error will be generated in the feedback file. The name of the feedback file will be created according to the following naming convention:

***FB\_TR\_ExternalFileName\_SEQ.XML***

<i>FB</i>	Indicates that it is a feedback file
<i>ExternalFileName</i>	Name on the uploaded transaction report
<i>SEQ</i>	A 4 digit sequence number. Must be unique per transaction file. Increments until file is accepted or AWE.
<i>TYPE</i>	Feedback files are always of the type XML

**Example:**

Name of the transaction report: *TR\_ExternalFileName.XML*  
Name of the feedback file; *FB\_TR\_ExternalFileName\_0001.XML*  
Name of the compressed feedback file; *FB\_TR\_ExternalFileName\_0001.ZIP*

*1.7.5.3 Feedback file content*

XML has been chosen as the file format for the feedback file. The file structure and error codes in the feedback file are defined in the XML-schema file “TransactionFeedback.xsd”. All fields and error codes in this structure are described in the document “Feedback from TRS Requirement specification” and “Error codes and error reason from TRS”.

*1.7.5.4 Expected actions*

Depending on the FileStatus of the Transaction report the reporting firm is expected to take different actions:

- **ACC** No action expected.



- **AWE** Correct the wrong transactions and send a new file with the corrected transactions and with a new sequence number in the file name.
- **REJ** Correct the errors and resend the file with the same file name.

Depending on the Status of the transactions in the Transaction report the reporting firm is expected to take different actions.

- **ACCEPTED** No action expected
- **FAILED** Correct the transaction and resend it in a new file together with other corrected transactions if any. The new file must have a new sequence number in the file name.
- **IGNORED** If the transaction has been sent before no action is expected. Otherwise apply a new TransactionReferenceNumber to the transaction and resend it in a new file together with other corrected transactions. The new file shall have a new sequence number in the file name.

#### **1.7.6 Client Information**

Client information submitted in the transaction report will be encrypted and stored separately in the database with high integrity and security. The information will only be accessible in plain text to authorised users at such time that transaction data are required for an investigation.



## **1.8 RF administration interface**

RF administration interface is a web application developed to give reporting firms information and a possibility to report information via HTTP.

The web site for reporting firms provides the following functions.

- **Registration of Mifid transactions**  
The web application gives the reporting firm a possibility to upload a transaction report file via HTTP or manually register a single transaction. If no transactions are made for one day the reporting firm can register an empty transaction file.
- **View history**  
Under History previously uploaded transaction reports can be viewed.
- **My account**  
The page shows the reporting firms contact information.
- **FAQ**  
Summary of the most frequently asked questions about TRS, the reporting process and the application functionality.
- **Contact information**  
Information about how to get in contact with the competent authority in case of any problems or questions.
- **Registration of Customer Client Information**  
When requested by the competent authority reporting firms will have to submit customer client information. The web application provides a way to enter single entries of customer client information but also a possibility to upload a file containing information about several customer clients.

### **1.8.1 Log in**

An administrator at a reporting firm who wishes to access TRS administrator functions must have an e-ID. He or she must also be registered in TRS in conjunction with the reporting firm via the web interface.

### **1.8.2 History and Statistics**

The administrator at the reporting firm can see status on all transferred files with information if there exists an error message that is connected to the file and number of errors connected to the file contents.

The administrator at the reporting firm has access to statistics and historical information from TRS for reported files. Examples of information:

- Number of transferred files and transactions per time period
- Transferred transactions containing errors, which must be corrected
- Number of errors in transferred files per time period

### **1.8.3 Account information**

The administrator at the reporting firm will have the possibility to alter the registered contact information. It is used by the competent authority to contact the reporting firm.



## 1.9 Venue Transaction report file (venue reporting only)

Venue transactions for reconciliation will only be required from and sent by a venue.

The content of the venue transaction report is based on MiFID and must adhere to given requirements regarding:

- File format (XML)
- Filename
- File type

### 1.9.1 File format (XML)

File format is defined as the XML structure that the venue transaction report must comply to. It is defined in the XML-schema file “VenueTransactionReport.xsd”. All fields in this structure are described in the document “TRS Report Format for venues”.

#### 1.9.1.1 Information type

There are two different information types in a venue transaction report file:

1. Transaction Record Info – transaction information
2. Cancellation Transaction Type – cancellation of a transaction

Venue transactions with the same value in the field “TransactionReferenceNumber” must only be present once for each information type (transaction or cancellation) in a single file.

### 1.9.2 Filename

The filename is according to following structure:

`RC_TRFID_YYYYMMDD_SEQ.TYPE`

Section	Explanation	Remarks
<i>RC</i>	Indicates that the file is a VenueTransaction report	
<i>TRFID</i>	BIC code for Technical Reporting firm	
<i>YYYYMMDD</i>	Reporting date that the transaction file refers to	
<i>SEQ</i>	Sequence number	A numerical 4 digits sequence number [0000-9999]. Must be unique per day.
<i>TYPE</i>	File type	See 1.9.3 for valid file types

#### Example:

Name on transaction report: `RC_FISSSEBORSX_20070601_0001.XML`



### 1.9.2.1 Incorrect filename

If a venue transaction report with an incorrect filename is reported the process flow of the file will be aborted and an error will be created in a feedback file

## 1.9.3 File type

Following file types are accepted by TRS.

File type	Interface	Remark
XML	✓	Se 1.9.3.1
ZIP	✓	Se 1.9.3.2
ZIP (ENC)	✓	Se 1.9.3.3

✓ = Accepted

### 1.9.3.1 XML file

The uploaded file must comply with the specified XML-schema. The content of the venue transaction report must be in a valid xml-format with Utf-8 encoding and it must be validated against the current xml schema before sending the file to the TRS system. XML-schema files are available on <http://schemas.fi.se/trs> for download.

Please note that only one namespace is allowed for TRS XML files. This means that the root element must consist of exactly two attributes: namespace and version. The prefix of the root element must correspond to the declared namespace. See example below:

```
<trs:report xmlns:trs="http://schemas.fi.se/TRS/InVenueTrans" Version="1.0">
```

### 1.9.3.2 Compressed files

The uploaded compressed file must comply with following:

- The file type must be ZIP and must only contain ONE compressed file (and no subfolders).
- The compressed file must be a XML file.
- The compressed file must use the same filename as the ZIP-file but with the ZIP extension.



### *1.9.3.3 Signed and encrypted files (ENC)*

Signed and encrypted files are created with a certificate. See 1.4 for information about how to sign and encryp files.

### **1.9.4 Feedback file**

The feedback file is the XML-file that will be returned to the venue after upload, validation and verification of the venue transaction file.

The returned feedback file will be compressed and stored in the “status” folder. If the venue transaction file was encrypted the feedback file will also be returned as encrypted.

Note: Encrypted feedback files must have been ordered from your competent authority and the venue transaction file must have either status “ACC” (Accepted) or “AWE” (AcceptedWithErrors) in order for the TRS system to return an encrypted feedback file.

The feedback files will be removed from the status folder after 10 days.

The venue transactions in the feedback file are identified by the transaction reference number of the submitted venue transaction. For cancellation venue transaction records, that do not have a transaction reference number, this number is the same reference number as the venue transaction referenced to be cancelled.

#### *1.9.4.1 Name on the feedback file – if the transaction report has a correct filename*

The name of the feedback file will be generated by the system and will reflect the name of the uploaded venue transaction file.

When a venue transaction report has been uploaded with a correct filename the returned feedback file will have the same name with prefix FB. The returned file will be compressed.

#### **Example:**

Name on the reported transaction file: *RC\_FISSSEBORSX\_20070601\_0001.XML*

Name on the associated feedback file: *FB\_RC\_FISSSEBORSX\_20070601\_0001\_0001.XML*

Name on the compressed feedback file: *FB\_RC\_FISSSEBORSX\_20070601\_0001\_0001.ZIP*

#### *1.9.4.2 Name on the feedback file – if the transaction file has an incorrect filename*

If the uploaded venue transaction file has an incorrect filename the processing flow of the file will be aborted and an error will be generated in the feedback file. The name of the feedback file will be created according to the following.

***FB\_RC\_ExternalFileName\_SEQ.XML***



<i>FB</i>	Indicates that it is a feedback file
<i>ExternalFileName</i>	Name on the uploaded transaction report
<i>SEQ</i>	A 4 digit sequence number. Must be unique per transaction file. Increments until file is accepted or AWE
<i>TYPE</i>	Feedback files are always of the type XML

**Example:**

Name of the venue transaction report;	<i>RC_ExternalFileName.XML</i>
Name of the feedback file;	<i>FB_RC_ExternalFileName_0001.XML</i>
Name of the compressed feedback file;	<i>FB_RC_ExternalFileName_0001.ZIP</i>

*1.9.4.3 Feedback file content*

XML has been chosen as the file format for the feedback file. The file structure and error codes in the feedback file are defined in the XML-schema file “VenueTransactionFeedback.xsd”. All fields and error codes in this structure are described in the document “Feedback from TRS Requirement specification” and “Error codes and error reason from TRS”.

*1.9.4.4 Expected actions*

Depending on the FileStatus of the Venue transaction report the venue is expected to take different actions:

- **ACC** No action expected.
- **AWE** Correct the wrong venue transactions and send a new file with the corrected venue transactions and with a new sequence number in the file name.
- **REJ** Correct the errors and resend the file with the same file name.

Depending on the Status of the venue transactions in the Venue transaction report the venue is expected to take different actions.

- **ACCEPTED** No action expected
- **FAILED** Correct the venue transaction and resend it in a new file together with other corrected venue transactions if any. The new file must have a new sequence number in the file name.
- **IGNORED** If the transaction has been sent before no action is expected. Otherwise apply a new TransactionReferenceNumber to the venue transaction and resend it in a new file together with other corrected venue transactions. The new file shall have a new sequence number in the file name.



## 1.10 Instrument file (venue reporting only)

Instrument information will only be required from and sent by a venue.

The content of the instrument report is based on MiFID and must adhere to given requirements regarding:

- File format (XML)
- Filename
- File type

### 1.10.1 File format (XML)

File format is defined as the XML structure that the instrument report must comply to. This is defined in the XML-schema files “InstrumentReport.xsd”. All fields in this structure are described in the document “TRS Report Format for instrument”.

#### 1.10.1.1 Information type

There is one information type in an instrument report file:

1. Instrument - Instrument information

#### 1.10.1.2 File content

Each reporting firm has the option to send a FULL or DELTA file. This means that each reporting firm has an attribute indicating if the content from the reporting party includes the full set of reference data for the reporting firm (F = Full) or if the content is only additions and modifications (D = Delta).

### 1.10.2 Filename

The filename is according to following structure:

IN\_TRFID\_YYYYMMDD\_SEQ.TYPE

Section	Explanation	Remarks
<i>IN</i>	Indicates that the file is an instrument file	
<i>TRFID</i>	BIC code for Technical Reporting firm	
<i>YYYYMMDD</i>	Reporting date that the instrument file refers to	
<i>SEQ</i>	Sequence number	A numerical 4 digits sequence number [0000-9999]. Must be unique per day.
<i>TYPE</i>	File type	See 1.10.3 for valid file



		types
--	--	-------

**Example:**

Name on the instrument file: *IN\_ExternalFileName.XML*

*1.10.2.1 Incorrect filename*

If an instrument report with an incorrect filename is reported the process flow of the file will be aborted and an error will be created in a feedback file.

**1.10.3 File type**

Following file types are accepted by TRS.

File type	Interface	Remark
XML	✓	Se 1.10.3.1
ZIP	✓	Se 1.10.3.2
ZIP (ENC)	✓	Se 1.10.3.3

✓ = Accepted

*1.10.3.1 XML file*

The uploaded XML file must comply with the specified XML-schema. The content of the instrument file must be in a valid xml-format with Utf-8 encoding and it must be validated against the current xml schema before the file is sent to the TRS system. XML-schema files are available on <http://schemas.fi.se/trs> for download.

Please note that only one namespace is allowed for TRS XML files. This means that the root element must consist of exactly two attributes: namespace and version. The prefix of the root element must correspond to the declared namespace. See example below:

```
<trs:report xmlns:trs="http://schemas.fi.se/TRS/InInstr" Version="3.0"
```

*1.10.3.2 Compressed files*

The uploaded compressed file must comply with following:

- The file type must be ZIP and must only contain ONE compressed file (and no subfolders).
- The compressed file must be a XML file.
- The compressed file must use the same filename as the ZIP-file but with the ZIP extension.



### *1.10.3.3 Signed and encrypted files (ENC)*

Signed and encrypted files are created with a certificate. See 1.4 for information about how to sign and encryp files.

### **1.10.4 Feedback file**

The feedback file is the XML-file that will be returned to the venue after upload, validation and verification of the instrument file.

The returned feedback file will be compressed and stored in the “status” folder. If the instrument file was encrypted the feedback file will also be returned as encrypted.

Note: Encrypted feedback files must have been ordered from the competent authority and the instrument file must have either status “ACC” (Accepted) or “AWE” (AcceptedWithErrors) in order for the TRS system to return an encrypted feedback file.

The feedback files will be removed from the status folder after 10 days.

The instruments in the instrument feedback file are identified by the ISIN code of the submitted instrument.

#### *1.10.4.1 Name on the feedback file – if the instrument file has a correct filename*

The name of the feedback file will be generated by the system and will reflect the name of the uploaded instrument file.

When an instrument file has been uploaded with a correct filename the returned feedback file will have the same name with prefix FB. The returned file will be compressed.

#### **Example:**

Name on the reported instrument file: *IN\_FISSSEBORSX\_20070601\_0001.XML*

Name on the associated feedback file: *FB\_IN\_FISSSEBORSX\_20070601\_0001\_0001.XML*

Name on the compressed feedback file: *FB\_IN\_FISSSEBORSX\_20070601\_0001\_0001.ZIP*

#### *1.10.4.2 Name of the feedback file – if the instrument file has an incorrect filename*

If the uploaded instrument file has an incorrect filename the processing flow of the file will be aborted and an error will be generated in the feedback file. The name of the feedback file will be created according to the following.

***FB\_IN\_ExternalFileName\_SEQ.XML***



<i>FB</i>	Indicates that it is a feedback file
<i>ExternalFileName</i>	Name on the uploaded instrument file
<i>SEQ</i>	A 4 digit sequence number. Must be unique per transaction file. Increments until file is accepted or AWE
<i>TYPE</i>	Feedback files are always of the type XML

**Example:**

Name on the instrument file:	<i>IN_ExternalFileName.XML</i>
Name on the feedback file;	<i>FB_IN_ExternalFileName_0001.XML</i>
Name on the compressed feedback file;	<i>FB_IN_ExternalFileName_0001.ZIP</i>

*1.10.4.3 Feedback file content*

XML has been chosen as the file format for the feedback file. The file structure and error codes in the feedback file are defined in the XML-schema "InstrumentFeedback.xsd". All fields and error codes in this structure are described in the document "Feedback from TRS Requirement specification" and "Error codes and error reason from TRS".

*1.10.4.4 Expected actions*

Depending on the FileStatus of the Instrument report the reporting venue is expected to take different actions:

- **ACC** No action expected.
- **AWE** Correct the wrong instrument entries and send a new file with the corrected entries and with a new sequence number in the file name.
- **REJ** Correct the errors and resend the file with the same file name.

Depending on the Status of the instrument entries in the Instrument report the reporting venue is expected to take different actions.

- **ACCEPTED** No action expected
- **FAILED** Correct the instrument and resend it in a new file together with other corrected instruments if any. The new file must have a new sequence number in the file name.



## 1.11 Customer Client Information reports

The content of the customer client report is based on MiFID and must adhere to given requirements regarding:

- File format (XML)
- Filename
- File type
- File size

### 1.11.1 File format (XML)

File format is defined as the XML structure that the customer client report must comply to. It is defined in the XML-schema files “CustomerClientReport.xsd”. All fields in this structure are described in the document “TRS Report Format for customer client information”.

#### 1.11.1.1 Information type

There is only one information type in a customer client report file:

1. CustomerClient - customer client information

Records with the same value in the field “Client” must only be present once for each information type in a single file.

### 1.11.2 Filename

The filename is according to following structure:

CC\_TRFID\_YYYYMMDD\_SEQ.TYPE

Section	Explanation	Remarks
<i>CC</i>	Indicates that the file is a Customer Client report	
<i>TRFID</i>	BIC code for Technical Reporting firm	
<i>YYYYMMDD</i>	Reporting date that the transaction file refers to	
<i>SEQ</i>	Sequence number	A numerical 4 digits sequence number [0000-9999]. Must be unique per day.
<i>TYPE</i>	File type	See 1.11.3 for valid file types

#### Example:

Name on Customer Client report: *CC\_FISSSEBANK1\_20070601\_0001.XML*



### 1.11.2.1 Incorrect filename

If a customer client report with an incorrect filename is reported the process flow of the file will be aborted and an error will be created in a feedback file.

### 1.11.3 File type

Depending on how the customer client report is -transferred -uploaded via HTTP or FTP the following file types are accepted.

File type	Interface		Remark
	HTTP	FTP	
XML	✓	✓	Se 1.11.3.1
ZIP	✓	✓	Se 1.11.3.2
ZIP (ENC)	✗	✓	Se 1.11.3.3

✓ = Accepted

#### 1.11.3.1 XML file

The uploaded XML file must comply with the specified XML-schema. The content of the customer client report must be in a valid xml-format with Utf-8 encoding and it must be validated against the current xml schema before the file is sent to the TRS system. XML-schema files are available on <http://schemas.fi.se/trs> for download.

Please note that only one namespace is allowed for TRS XML files. This means that the root element must consist of exactly two attributes: namespace and version. The prefix of the root element must correspond to the declared namespace. See example below:

```
<trs:report xmlns:trs="http://schemas.fi.se/TRS/InClient" Version="1.0">
```

#### 1.11.3.2 Compressed files

The uploaded compressed file must comply with following:

- The file type must be ZIP and must only contain ONE compressed file (and no subfolders).
- The compressed file must be a XML file.
- The compressed file must use the same filename as the ZIP-file but with the ZIP extension.



### 1.11.3.3 Signed and encrypted files (ENC)

Signed and encrypted files are created with a certificate. See 1.4 for information about how to sign and encrypt files.

### 1.11.3.4 File size

Depending on if the customer client file is uploaded via HTTP or FTP there is size restrictions:

File type	Interface	
	HTTP	FTP
XML	35 Mb	No limit
ZIP	200 Kb	No limit
ZIP (ENC)	✘	No limit

If a customer client file is too large when uploaded via http the process flow of the file will be aborted and an error will be generated in the feedback file.

### 1.11.4 Feedback file

The feedback file is the XML-file that will be returned to the Technical Reporting Firm after upload, validation and verification of the customer client file.

The returned feedback file will be compressed and stored in the “status” folder. If the customer client file was encrypted the feedback file will also be returned as encrypted.

Note: Encrypted feedback files must have been ordered from your competent authority and the customer client file must have either status “ACC” (Accepted) or “AWE” (AcceptedWithErrors) in order for the TRS system to return an encrypted feedback file.

The feedback files will be removed from the status folder after 10 days.

The customer client records in the feedback file are identified by the customer client reference number of the submitted report.

#### 1.11.4.1 Name of the feedback file – if the customer clients file has a correct filename

The name of the feedback file will be generated by the system and will reflect the name of the uploaded customer client file.

When a customer client report has been uploaded with a correct filename the returned feedback file will have the same name with prefix FB. The returned file will be compressed.

#### **Example:**

Name on the customer clients file: *CC\_FISSSEBORSX\_20070601\_0001.XML*

Name on the associated feedback file: *FB\_CC\_FISSSEBORSX\_20070601\_0001\_0001.XML*

Name on the compressed feedback file: *FB\_CC\_FISSSEBORSX\_20070601\_0001\_0001.ZIP*



#### 1.11.4.2 Name of the feedback file – if the customer client file has an incorrect filename

If the uploaded customer client file has an incorrect filename the process flow of the file will be aborted and an error will be generated in the feedback file. The name of the feedback file will be created according to the following.

#### ***FB\_CC\_ExternalFileName\_SEQ.XML***

<i>FB</i>	Indicates that it is a feedback file
<i>ExternalFileName</i>	Name on the uploaded customer client file
<i>SEQ</i>	A 4 digit sequence number. Must be unique per transaction file. Increments until file is accepted or AWE
<i>TYPE</i>	Feedback files are always of the type XML

#### **Example:**

Name of the Customer Client report:	<i>CC_ExternalFileName.XML</i>
Name of the feedback file;	<i>FB_CC_ExternalFileName_0001.XML</i>
Name of the compressed feedback file;	<i>FB_CC_ExternalFileName_0001.ZIP</i>

#### 1.11.4.3 Feedback file content

XML has been chosen as the file format for the feedback file. The file structure and error codes in the feedback file are defined in the XML-schema file “CustomerClientFeedback.xsd”. All fields and error codes in this structure are described in the document “Feedback from TRS Requirement specification” and “Error codes and error reason from TRS”.

#### 1.11.4.4 Expected actions

Depending on the FileStatus of the Customer client report the reporting firm is expected to take different actions:

- **ACC** No action expected.
- **AWE** Correct the wrong customer client entry and send a new file with the corrected customer client entries and with a new sequence number in the file name.
- **REJ** Correct the errors and resend the file with the same file name.

Depending on the Status of the customer client entries in the Customer client report the reporting firm is expected to take different actions.

- **ACCEPTED** No action expected

Title:  
**TRS Requirement  
specification**  
doc type  
**Requirement specification 3.0**

Created:  
**8/29/2008 11:12:00 AM**  
Last changed on:  
**3/16/2011 3:00:00 PM**

Author:  
**TRS**  
Last changed by:  
**TRS**



- 
- **FAILED** Correct the customer client entry and resend it in a new file together with other corrected customer client entries if any. The new file must have a new sequence number in the file name.



## 1.12 Transaction report file (OTC transactions)

The content of the OTC transaction report is based on MiFID and must adhere to given requirements regarding:

- File format (Valid XML)
- Filename
- File type
- File size

Transactions from several Reporting Parties can be sent together in one file by a Technical Reporting Party or in one file for each Reporting Party, whichever is preferred.

### 1.12.1 File format (XML)

File format is defined as the XML structure that the transaction report must comply to. It is defined in the XML-schema file “OTCTransactionReport.xsd”. This file also includes the files “TRSDataTypes.xsd”, “StandardDataTypes1.0.xsd” and “TREM\_TransactionRecordTypes3.0.xsd”. All fields in OTCTransactionReport.xsd are described in the document “TRS Report Format for reporting party (OTC)”.

#### 1.12.1.1 Information type

There are two different information types in a transaction report file:

1. OTCTransaction Record Info - transaction information
2. Cancellation Transaction Type – cancellation of a transaction

Transactions with the same value in the field “TransactionReferenceNumber” must only be present once for each information type (transaction or cancellation) in a single file.

### 1.12.2 Filename

The filename is according to following structure:

OT\_TRFID\_YYYYMMDD\_SEQ.TYPE

Section	Explanation	Remarks
<i>OT</i>	Indicates that the file is a Transaction report	
<i>TRFID</i>	BIC code for Technical Reporting firm	
<i>YYYYMMDD</i>	Reporting date that the transaction file refers to	
<i>SEQ</i>	Sequence number	A numerical 4 digits sequence number [0000-9999]. Must be unique per day.



<i>TYPE</i>	File type	See 1.12.3 for valid file types
-------------	-----------	---------------------------------

**Example:**

Name on transaction report: *OT\_FISSSEBANK1\_20110601\_0001.XML*

### 1.12.2.1 Incorrect filename

If a transaction report with an incorrect filename is reported the process flow of the file will be aborted and an error will be created in a feedback file.

## 1.12.3 File type

The following file types are accepted.

	Interface	Remark
<b>File type</b>	<b>FTP</b>	
XML	✓	Se 1.12.3.1
ZIP	✓	Se 1.12.3.2
ZIP (ENC)	✓	Se 1.12.3.3

✓ = Accepted

### 1.12.3.1 XML file

The uploaded XML file must comply with the specified XML-schema. The content of the transaction report must be in a valid xml-format with Utf-8 encoding and it must be validated against the current xml schema before the file is sent to the TRS system. XML-schema files are available on <http://schemas.fi.se/trs> for download.

Please note that in this schema no namespace is used, this is a difference to the other TRS XML file types.

### 1.12.3.2 Compressed files

The uploaded compressed file must comply with following:

- The file type must be ZIP and must only contain ONE compressed file (and no subfolders).
- The compressed file must be a XML file.
- The compressed file must use the same filename as the ZIP-file but with the ZIP extension.



### *1.12.3.3 Signed and encrypted files (ENC)*

Signed and encrypted files are created with a certificate. See 1.4 for information about how to sign and encryp files.

### **1.12.4 Feedback file**

The feedback file is the XML-file that will be returned to the Technical Reporting Firm after upload, validation and verification of the transaction file.

The returned feedback file will be compressed and stored in the “status” folder. If the transaction file was encrypted the feedback file will also be returned as encrypted.

Note: Encrypted feedback files must have been ordered from your competent authority and the transaction file must have either status “ACC” (Accepted) or “AWE” (AcceptedWithErrors) in order for the TRS system to return an encrypted feedback file..

The feedback files will be removed from the status folder after 10 days.

The transactions in the feedback file are identified by the transaction reference number of the submitted transaction. For cancellation transaction records, that do not have a transaction reference number, this number is the same reference number as the transaction referenced to be cancelled.

#### *1.12.4.1 Name of the feedback file – if the transaction report has a correct filename*

The name of the feedback file will be generated by the system and will reflect the name of the uploaded transaction file.

When a transaction report has been uploaded with a correct filename the returned feedback file will have the same name with prefix FB. The returned file will be compressed.

#### **Example:**

Name on the reported transaction file: *OT\_FISSSEBANK1\_20070601\_0001.XML*

Name on the associated feedback file: *FB\_OT\_FISSSEBANK1\_20070601\_0001\_0001.XML*

Name on the compressed feedback file: *FB\_OT\_FISSSEBANK1\_20070601\_0001\_0001.ZIP*

#### *1.12.4.2 Name of the feedback file – if the transaction file has an incorrect filename*

If the uploaded transaction file has an incorrect filename the transaction flow will be aborted and an error will be generated in the feedback file. The name of the feedback file will be created according to the following naming convention:

***FB\_OT\_ExternalFileName\_SEQ.XML***



<i>FB</i>	Indicates that it is a feedback file
<i>ExternalFileName</i>	Name on the uploaded transaction report
<i>SEQ</i>	A 4 digit sequence number. Must be unique per transaction file. Increments until file is accepted or AWE.
<i>TYPE</i>	Feedback files are always of the type XML

**Example:**

Name of the transaction report;	<i>OT_ExternalFileName.XML</i>
Name of the feedback file;	<i>FB_OT_ExternalFileName_0001.XML</i>
Name of the compressed feedback file;	<i>FB_OT_ExternalFileName_0001.ZIP</i>

*1.12.4.3 Feedback file content*

XML has been chosen as the file format for the feedback file. The file structure and error codes in the feedback file are defined in the XML-schema file "TransactionFeedback.xsd". All fields and error codes in this structure are described in the document "Feedback from TRS Requirement specification" and "Error codes and error reason from TRS".

*1.12.4.4 Expected actions*

Depending on the FileStatus of the Transaction report the reporting firm is expected to take different actions:

- **ACC** No action expected.
- **AWE** Correct the wrong transactions and send a new file with the corrected transactions and with a new sequence number in the file name.
- **REJ** Correct the errors and resend the file with the same file name.

Depending on the Status of the transactions in the Transaction report the reporting firm is expected to take different actions.

- **ACCEPTED** No action expected
- **FAILED** Correct the transaction and resend it in a new file together with other corrected transactions if any. The new file must have a new sequence number in the file name.
- **IGNORED** If the transaction has been sent before no action is expected. Otherwise apply a new TransactionReferenceNumber to the transaction and resend it in a new file together with other corrected transactions. The new file shall have a new sequence number in the file name.

Title:  
**TRS Requirement  
specification**

doc type

**Requirement specification 3.0**

Created:  
**8/29/2008 11:12:00 AM**

Last changed on:  
**3/16/2011 3:00:00 PM**

Author:  
**TRS**

Last changed by:  
**TRS**



---

### **1.12.5 Client Information**

Client information submitted in the transaction report will be encrypted and stored separately in the database with high integrity and security. The information will only be accessible in plain text to authorised users at such time that transaction data are required for an investigation.