Crosskey Banking Solutions
PKI File Transfer Technical Description for software supplier
Version 2.4.2, January 2025

Contents

1 Revision list

Most recent change	Changed by	What was changed

2 Introduction

This specification should be seen as an addition to the following documents from the bank association:

"Security and Message Specification For Financial Messages Using Web Services".
BankCorporateFileService_20080616.wsdl
ApplicationRequest_20080918.xsd
ApplicationResponse_20080918.xsd

They can be found through searching for WebServices on www.fkl.fi

The services from the specification implemented in step one are UploadFile, DownloadFileList, DownloadFile and GetUserInfo.

Enrolling for the service is done in 2 steps. Step one takes place at the bank-office and step 2 takes place when the customer is back at the company office and subscribes through his Corporate bank software. When the customer has typed in necessary information into the bank-connection program, the software does a web-service call to the bank and the service getCertificate. A central piece of the information needed here is the Transfer key and this is delivered to the customer in 2 parts.

3 Description service by service

3.1 GetCertificate

The service is described in CKCertService.wsdl, CertApplicationRequest.xsd, CertApplicationResponse.xsd.

The three most important inparameters here is the public key, customerId and transferkey.
The public key and customerId are both part of the PKCS10-package. The customerId should be in CNstring as this:

CN=[CustomerId up to 10 digits, no leading zeros]

The transfer key comes in CertApplicationRequest. Transfer key is 16 digits. The algorithm used here to create a verification digit is Luhn:

http://en.wikipedia.org/wiki/Luhn_algorithm

The last digit is the checksum digit.

In production the field Environment has to be PRODUCTION otherwise the request will be rejected.

CertApplicationRequest and SOAP don't need to be signed on the registration-call.

3.1.1 Certificate renewal

On certificate renewal, a signed CertApplicationRequest is used and the Transfer key should be empty. The Certificate used in the signature is the certificate that is expiring and this only works during the renewal period. Once the certificate is expired new shared secrets have to be requested from the bank. The renewal period is:

- 60 days in test, where the total validity time is 61 days, meaning that it is almost the whole period.
- In production the renewal period is 60 days before the certificate expires. New private key and csr should be generated in the renewal scenario and not only a new request with the old csr, thus you can also call this renewal rekeying.

3.1.2 Keys and algorithms.

Private key should be a RSA key, 2048 bytes long. Material should be signed using SHA1 or SHA256. Certificate request should be in PEM form and then binary64-encoded into the Certificate Applicationrequest. This mean that if you have the key in DER-form it should be binary64-encoded once, placed in between "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----", then binary-64 encoded again and the bin64-string then placed in content.

3.1.3 Certificate revocation list

When certificates are revoked, their serial numbers are placed in the CRL-list which then is publiched on the following LDAP-URL:s

POP-Pankki test CA
[1]CRL Distribution Point
URL=http://httpcrl.trust.telia.com/poptestcav1.crl

Verification of that a certificate isn't on the CRL is preferably done each time a signature is verified, meaning that on client side this should be done when they verify the validity of the banks response messages.

3.2 UploadFile

The materials are sent one by one through UploadFile.
We prefer if the content in applicationrequest is compressed, in particular for large files.
The compression method should be GZIP and compression of content should be made before base64encoding of content. In the ApplicationRequest, compression should be set to true when compression is used.

We do not support encryption of the Applicationrequest content, so Applicationrequest encryption should be set to false.

3.3 DownloadFileList

This service is used to retrieve a list of files in the bank. Fetched materials can be fetched again through specifying status DLD or ALL. With status DLD (downloaded) fetched materials will be listed. With status ALL, all materials regardless of status will be listed.

3.4 DownloadFile

The materials are fetched one by one through DownloadFile. This means that we only support one file reference in one request. The unique file reference fetched through DownloadFileList has to be specified in the field FileReference. For realtimebalances (0710) and transactions (0711) please see chapter 4.

In order for the downloaded files to be compressed, *<Compression>true</Compression>* must be set in the ApplicationRequest.

GetUserInfo

This service will retrieve the file types the user has access to.

4 Realtimebalances and transactions(0710 and 0711)

Real-time balances and transactions can be downloaded either through a direct call to DownloadFile or by first calling DownloadFileList. When calling DownloadFileList, the result will return a list of connected accounts.

For direct download without calling DownloadFileList, instead of using the unique file ID as a file reference, you can specify the account number directly in the FileReference field as shown below:

```
<FileType>(fileType: 0710 or 0711)</FileType>
<FileReferences>
   <FileReference>FI5366010005634415</FileReference>
   <FileReference>FI5366010005634416</FileReference>
</FileReferences>
```

Multiple accounts per request are supported, with each account number in the request formatted as IBAN. A file will be generated on-the-fly and returned in the content of the application response as usual via **DownloadFile**.

The maximum number of accounts per request is 50. If the request contains more than 50 accounts, the request will be denied**.**

5 Response codes

Code	Name	Remarks
00	Ok	Everything went as it should
02	SOAP signature error	Soap-level signature verification failed
07	Contract not valid	Certificate has expired, is faulty or other reason that fails authentication and authorization.
12	Schema validation failed	To large file, SOAP envelope or body validation error, Schema validation error
25	Content not allowed	Content not allowed
26	Technical error	Examples: Backend query failed, result was null or empty when a value was expected.
29	Invalid parameters	If request parameters were null or empty.

6 File type codes

The application codes currently used with PATU will not change when switching to PKI. In the list below the file type codes available with PKI are displayed.

POP-Pankki

Code	Name**	Explanation
0100	SCCT	CustomerCreditTransferInitiation
0130	SPSR	CustomerPaymentStatusReport
0250	REFG	Incoming reference payments
0391	VALK	Currency
0410	SKÅT	IBAN control & conversion
0420	SKÅR	IBAN control & conversion return
0700	KTOF	Account statement
0702	KNCT	Group statement + transactions
0710	SALD	Balances
0711	TRAN	Balances + transactions
0800	EFAK	E-invoice sending
0810	EFAS	E-invoice return answer
0820	EFAH	E-invoice fetching
0830	ESAN	Sender Info
0840	EMOT	Receiver info
0850	EFRP	Receiver proposal
0860	EFOS	E-invoice return for operator
1250	REFX	Reference transactions XML
1700	KTOX	Account statement XML
1702	KNCX	Group statement + transactions XML
1710	SALX	Balances XML
1711	TRAX	Balances + transactions XML
2100	MYIN	Authority requests
2105	MYUT	Authority requests – return material

**= Name for internal use only.

7 Service url:s

CertificateService

Environment	Bank	Adresses
Production	POP Pankki	https://pft.poppankki.fi/CertificateService-v1
Stage	POP Pankki	https://popstagepft.crosskey.fi/CertificateService-v1

CorporateFileService

Environment	Bank	Adresses
Production	POP Pankki	https://pft.poppankki.fi/CorporateFileService-v1
Stage	POP Pankki	https://popstagepft.crosskey.fi/CorporateFileService-v1

8 Charset for the applications

There will be no change in the character sets used for the applications. That means that SEPA files use UTF-8 and all other filetypes use ISO8859-1.

9 Other information

9.1 Test

Test is done in test environment. We do not support test in production environment thru specifying the xml-element environment to 'Test'. In production the environment has to be 'Production'.

10 Root Certificates

The root certificates are provided here and they are also available on the LDAP where the CRL-list resides.

10.1 POP-Pankki test

-----BEGIN CERTIFICATE-----
MIIFcTCCA1mgAwIBAgIPAYPWJ0OqbBqeTicvJ9QNMA0GCSqGSIb3DQEBDAUAMEEx
CzAJBgNVBAYTAkZJMRkwFwYDVQQKDBBQT1AgUGFua2tpa2Vza3VzMRcwFQYDVQQD
DA5QT1AgVGVzdCBDQSB2MTAeFw0yMjEwMTQxMTAxMDhaFw00NzEwMTQxMTAxM
Dha
MEExCzAJBgNVBAYTAkZJMRkwFwYDVQQKDBBQT1AgUGFua2tpa2Vza3VzMRcwFQYD
VQQDDA5QT1AgVGVzdCBDQSB2MTCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoC
ggIBAJuluYyx8jZ1GYtUb5sWp3xuqGwhrEEYETDsMSJT37GH9QXE2UniEThR5Y0J
hLdxOsaJLnqT4k40TVzE+22BB1nGXxVOsInb9pgKpe7p+TTZ7SwFiohPkudm3vQm
cn8MjWHTvoZ+0S9s4z9K22eLYi6V54jObdmjNnzy/hILLtBZ0W1xdg4cV71l6PWt
c8Gk5ILWm5ECZ2Z3zglOz8JngX1r5U3AK/gO7a/vcFI2ZSQCFSZZ75jR6U7KI7te
dlDDHGPI01pSHElDEarSss9E8yVP0BSepIpXpFvZUhTTsE7AZEzb2Na9C7tw49OK
UTW1SV0T7t1HC6uL+nBb3ixE/ZoTY8wGlqJmw99jfLi36+pSPrMSU5BFzbkC6YVb
OJk6dPqmrtkBYhG39rxWVhW92JHSMk84Z3HNS2RyI02RDIXmPBhlsGX+lDmdkeg9
kcRs5/D+5stdUYzcuPjuVbNNK1JmDAoIL11F5K6MPiiqywXgDbjZu4NmhQoZ/CbN
qLX988Q1Bom1+ebK+ql7t4N8p8pqdUfLiS/1OPrhfJ5jjrfYjEwdWILKOCpxyiJg
FPh+esHSoVkL8AZ4iI5zpM+zMrvkSvfk7SmZiBMJgkaOrdygpHhDFVxPfc2Pox32
lfJFAXzUoxPLo8bP80Vm8YPJ7Zxfa3pp77QPHaIvuF4vWL/zAgMBAAGjZjBkMB8G
A1UdIwQYMBaAFGk+0ASxrMRkFmZzEgWwq419U0zWMB0GA1UdDgQWBBRpPtAEsazE
ZBZmcxIFsKuNfVNM1jAOBgNVHQ8BAf8EBAMCAQYwEgYDVR0TAQH/BAgwBgEB/wIB
ADANBgkqhkiG9w0BAQwFAAOCAgEAgGI2B4ks3UyIPa3Xl+RF0OHZN2JBUpz3/WS2
1vcS010sUKgjkHdDoGfDFYDAsXIgRJ6aOX/tCxhzPBMC66YefrMV+TbUyaf0ucBu
Vf7FqqICH/ENMXy2sd+9SudOhsDMX3Q2cIp3fBty6ILoXNw6mIZJ5aW7fuI3WK3y5
VYl1a+pup6hUHttf0XrO4UtpWd4To+hAkFw8W7ybDWsab7XViVkO78kSJnDJdg9M
-----

rMXbPAVP01PJuz0QaYYYIEyDumDGTW+bryefiyZya+yJV7HWnU+5l/n8Eh+V6A9w
0bQJLsPT04WzgKi8VinAAr9ep1O1xZ8TizueFrC6o5kEDZeC0BtHolBxiQoh5CxZ
bGkHkkp+5DUw7afZFktbISQymsEpCs094TqMDXGUM3j+Qpo6gboKUV2V/+Aaj05C
lPUrFo6KeiQcaYe8CISm2Jh/UKmwD0Vj97bJVoF9og8QrPO9LYFlupDrjJdHaN6i
fnaqqC3IG/msSUNVmx/hZhg0L/hmh8PI7O7EWPVmPAX3pdCGZF9ZnnG8P9UQObUP
2oWQd/Pvw4b3yuzZYit0CqG8JBQ67cHP4oLfbgROQsPMrgx+FtX7aKrzUNohOgNq
EW8HqZBmQnwQmP0hpHJ6uF1knLfW3BlfV4osTQ0+WT+YZ7rkFBnhz3BPF+m1aFyS
OcdqTd0=
-----END CERTIFICATE-----

10.2 POP-Pankki production

-----BEGIN CERTIFICATE-----
MIIFZzCCA0+gAwIBAgIPAZM+iWWqjZOZzZscl6qfMA0GCSqGSIb3DQEBDAUAMDwxCzAJBgNVBAYT
AkZJMRkwFwYDVQQKDBBQT1AgUGFua2tpa2Vza3VzMRIwEAYDVQQDDAlQT1AgQ0EgdjEwHhcNMjQx
MTE4MDkxMDI4WhcNNDkxMTE4MDkxMDI4WjA8MQswCQYDVQQGEwJGSTEZMBcGA1UECgwQUE9QIFBh
bmtraWtlc2t1czESMBAGA1UEAwwJUE9QIENBIHYxMIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIIC
CgKCAgEAn86jy9fcKwNQ4QQwLXVGtZfR00ILWuub3/vgGKliRRRKKiGfBq2ltMfcUNLEZJv+N1I9
PMX8xJ+y9mArE/EhTt3Pt/FtVWJLdUsjZZcnQVw04zrzQH+IMpTEyK3iTPWMkb6dlS7VeHBhRokS
pRkjYxyF3EN/qBbOERXNGMKUcltJMVuAJ/lwbvptRgQZ4ZKdLqlnPxFbDApVTVIzcoISSuJlNY0F
o+yUO7L5tdpyyVn0iB+flcUiccZByHm8XVESoxl1juSZRyKduJdZLlfcf7j+XGAYdqkOSGPJpXh6
darxFFXzC1B1r7qcKsDCMH6UG5m/KyA7BcZl/9af2lXT/b7h98EWXW+4jqyw+uY4+eh8UJF/g0Rm
dcgPk32fDo+JL1pIRP9xKejlQn6KwhUyNMDG4DJGOb7tSjUdkWrDT6FZelblm4pQfMQ/dTfzCaS+
dU4MfM8ZE6mN1zr6gcJIxQjN2rWby/sCDFS3nz0av2w7UQ367jZSaQ81beonuBdFIzKIhhbJ9BE6
xpH1WSQ/+5n8Skp2VjiQ3SfxjumUZ9U2h5yQrEcy6GmKvnU6wfgNaaaXt5IfyeRTYuCWeWZ7gFMk
EnlU9NPdmWuqwLJ2Yge/2ty8323wz8HtoSluQd31fCH0JGzxWJnwKd2HGpAs2EEk2cUAnUmuMhfb
444cOQUCAwEAAaNmMGQwHwYDVR0jBBgwFoAUZ7LO251HyYJZWvue0XRIOvIyRN8wHQYDVR0OBBYE
FGeyztudR8mCWVr7ntF0SDryMkTfMA4GA1UdDwEB/wQEAwIBBjASBgNVHRMBAf8ECDAGAQH/AgEA
MA0GCSqGSIb3DQEBDAUAA4ICAQA3FBgFks/2RhpHRdD0l82fiX4Fr7O3oClD9+gqOtQse7Uug2G0
A3T3tM3BVd3GuAX9TfkE+FePJ1O6sbpfmLxA4QGiTcwtQfX/Z+5QCFn7mlGvg0xB7vPxLZOtmpNh
5TrGWi3hEfhCjwcXWqU71lniC/Go7p+AEXUO4Q///mzzfViaiDtP2bryQ6AEL5QDboSF1Geile5l
PXAFRuNg5So+rixZb0Sn5pMgcTluYbm4eIdc4nIZAdn7vg6METXXh7K9unjgwb1Imp4/zhTOR6Nr
xsremRoCsLYC2lI+0gKhkcgscEJ1YajfijdjDkZ+ysi1kRbjhHIqeYcmiatttPw7CYQSsD3JlA4Y
/SS8fuNtH8PqsXocBh+cidAlpa1+NVC1Vy0CeUMxrtxAopyXf1ZQF/QlYsqAtbMQk2Yy3FwX2SWx
5vvyQ8wTtUaCnsVUFBT3i/a7BN8id5URuURblL+Y1Y6cld1KGfw1rL0+ZAmd8RmCsLuMNWlYELMO
yNdCypECZukg6CNcsclBJT2hAWhhvQPru20wY1NZmXZcJryUMNLSO/1nApk6eTbeuW3CX906oK6+6
9gLbQuj/+ZkhlCOJjpvpRB21f1obo33w3LsUUBu4W/HQNmAIAPyc0KCnkyW3nj8KF9v2L0HGV9bY
0m6ABv5HBO7mY9n7wJPN029geA==
-----END CERTIFICATE-----