EBICS examples

Management Summary

The following examples illustrate the inner workings of the EBICS specification. They will concentrate on the cryptograhpic aspects – that is the canonicalization of the XML, calculation of hash codes and signatures, encryption and decryption.
To achieve the final goal of sending a file via EBICS, keys have to be exchanged between client and server. The key exchange is part of this example.
The actual data transmitted is of less importance for this example. We therefore transmit some arbitrary data without any bank related meaning – we will use the EBICS logo for this.
All the intermediate steps are stored in files which we provide together with this example.
Two examples will be provided. The first is according to the EBICS specification as it used in Germany – that is, keys are used for INI and HIA, and the actual upload is done as a generic upload using an order type agreed between user and bank.
The second example relates to the French market and therefore uses certificates instead of keys. The upload is done with an FUL transaction, the file format is again arbitrarily chosen.

Workflow

The general wokflow is the same for both examples:

1. The user is configured in the bank system
2. The bank parameters are sent to the user
3. The user creates his private keys (and the certificates if necessary)
4. The user sends a INI request to publish his A005 key
5. The users prints, signs and sends a INI letter for this key
6. The user sends a HIA request to publish his E002 and X002 keys
7. The users prints, signs and sends a HIA letter for these keys
8. The bank verifies the keys by the INI and HIA letter and activates the user
9. The user sends an HPB request to download the bank's keys
10. The user verifies the bank's keys
11. The user sends the actual data

From then on, all uploads and downloads are just a repetition of the last step.

Preliminaries

The following rules apply to the way binary or numerical data is printed throughout this example:

- Hash codes are printed hexadecimal
- Big numbers (e.g. modulus or exponent of RSA keys) are printed as big-endian hexadecimal numbers
- Certificates are printed in PEM encoding
- XML is printed without any additional formatting but longer lines will be broken. XML files with the original content are provided along with this example.

Bank Parameters

To compile this example, we use a bank server. reachable under https://194.180.18.30/ebicsweb/ebicsweb, the host id is SIZBN001. For the first example, we define a new partner with the partner id EBICS and a user with the used id EBIX. For the second example, a second partner with the id EBIXCERT and a user again with the id EBIX. The bank system uses the same key for encryption and authentication. The SHA-256 hash value of this key is:

```
74 50 18 7B 6F 35 BE 3F 4D 07 BC 3E 56 85 88 75
F1 E7 8D 8F 61 35 B6 4B 6C 7B 03 3A EE FA 40 01
```

User keys and certificates

The user generates three new key pairs: one for authoristation, one for authentication and one for encryption. A certificate is created for each key. The keys and certificates are:

A005

```
Modulus:
D7 8E 68 ED 9F 1E 5E 7A 6B B6 DC 4B 81 40 9D F4
F2 BC 68 A2 6E 68 B2 79 DF 49 C7 5C 22 7C 2A 23
BB 3C CB A6 74 95 5A 76 C3 9B 6C 32 07 5F D8 5C
AD 55 FD C9 65 2B E2 C2 AD AB F8 F3 13 27 A2 06
B4 69 17 15 C6 B4 82 B6 90 16 F8 F0 7A 5A 7D 61
2A 43 56 DA 6F F0 22 E3 F5 56 0F 8B 07 61 00 B0
05 6F 0A 23 2B 0A 9C 86 29 45 06 35 0E 71 D0 F8
7D D7 7C 58 52 06 78 D5 1A BF 08 D2 76 C3 80 2D
B8 28 1B AF 92 AF 45 3C 2F 28 4B F8 24 42 79 AB
29 9E 76 AD D8 CB 59 2D EA DE 47 6D 09 0B CC CA
4C 38 1B 4C 2F 35 01 89 95 2F D6 C4 C0 A4 4B 4A
DF D0 9D 08 8F 06 00 4B 19 75 6E 5E 3D 27 6B EF
BC B4 AA C0 3B 40 24 F1 5A 58 8B A4 D0 C1 C7 80
4E B9 82 73 F4 87 5E 8E AB 4D 4F 8C C0 0F BB 10
07 A6 1A 8B 9E 0B CC 9D 25 A4 79 80 DD B6 8A DF
F6 37 80 12 6A C4 2D D6 1D 20 5F EB CC 7A EA 27


Public exponent:
01 00 01


SHA-256 digest:
F5 AC B7 B5 CF 88 DC C8 09 05 AA E8 78 3E D7 25
F3 AD 1D CA BB 21 1D B7 7E 58 D6 79 F9 74 77 39


Private exponent:
29 78 D3 C5 4C 04 EA C6 80 EC D8 AB 8A 3D A5 66
```

```
58 41 59 9C 4E C3 C3 FA E0 B3 F0 30 50 CE 34 C9
55 1F D8 4A B9 49 76 F3 3C A5 2D 86 DE 96 59 29
53 8C 24 DC D7 A3 3E E8 97 C3 6B 8D 50 D5 22 3E
49 FD A2 1C 65 73 9B 66 86 88 74 F2 C7 87 9B 71
D3 50 CB 68 11 7B 51 2B 2C AB 97 27 F7 8E 79 34
64 87 3D 9B 10 8F E5 17 86 DF 29 D7 91 64 E2 80
BC AD 75 D6 04 4A 0C 7B 6B BE EF 72 96 B8 FC 72
F4 F5 01 24 AD 90 86 F7 69 B8 81 AE 3F AE FC 12
47 F5 7F 51 CC 0D 6A B7 FD EB F5 B8 95 E2 A5 9A
07 D7 BB 3D 15 EC C7 7A B5 FA 2E 6D CD E9 9D F2
3C 23 D0 FB 5E 9F 5E 45 1C 56 2E A2 CD 86 92 79
47 92 49 3A C0 BC A8 8C 2E BC DF F6 AA 3B 53 F0
29 40 CE 0F 39 0D 2B BB 85 4C 6F F1 C5 6A C5 C5
9E 8F 2F A1 62 99 E7 A2 50 DA 4E DD BD 1E D8 39
42 8D 3E 2B A0 CC 68 10 3F A2 2B 0F B3 3D AD E1


Certificate:
-----BEGIN CERTIFICATE-----
MIIDQDCCAiigAwIBAgIBATANBgkqhkiG9w0BAQsFADBOMQswCQYDVQQGEwJGUjEOMAwGA1UEBxMF
UGFyaXMxEzARBgNVBAoTCkUuQi5JLkMuUy4xCTAHBgNVBAsTADEPMA0GA1UEAxMGRS4gQml4MB4X
DTA5MDcyNzA5MDYxNVoXDTE0MDcyNzA5MDYxNVowTjELMAkGA1UEBhMCRlIxDjAMBgNVBAcTBVBh
cmlzMRMwEQYDVQQKEwpFLkIuSS5DLlMuMQkwBwYDVQQLEwAxDzANBgNVBAMTBkUuIEJpeDCCASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANeOaO2fHl56a7bcS4FAnfTyvGiibmiyed9Jx1wi
fCojuzzLpnSVWnbDm2wyB1/YXK1V/cllK+LCrav48xMnoga0aRcVxrSCtpAW+PB6Wn1hKkNW2m/w
IuP1Vg+LB2EAsAVvCiMrCpyGKUUGNQ5x0Ph913xYUgZ41Rq/CNJ2w4AtuCgbr5KvRTwvKEv4JEJ5
qymedq3Yy1kt6t5HbQkLzMpMOBtMLzUBiZUv1sTApEtK39CdCI8GAEsZdW5ePSdr77y0qsA7QCTx
WliLpNDBx4BOuYJz9IdejqtNT4zAD7sQB6Yai54LzJ0lpHmA3baK3/Y3gBJqxC3WHSBf68x66icC
AwEAAaMpMCcwCwYDVR0PBAQDAgZAMAoGA1UdDgQDBAExMAwGA1UdIwQFMAOAATEwDQYJKoZIhvcN
AQELBQADggEBALo97IHqI3SfAS7rcTRGykCxqImtaDKJ3c77iKaUBIfJtBWuSxMLbLyC5xG5WQmS
/IdjSLp8WWetDylP2mkKvqk601UJGYHTnvUHblVY9MhG5kD6LB6bfjp6PfOm1629Mlmg5BCZ7cO1
+pEbQK2IE1MTreCYPiQyJJeT6umUKfeRgS3WFWCya/zfaLlCZDrRy+N8TVFKauMMKa8AugUn9eAm
7CuKgt89RkH3ERhJUZWb5Rbo6ykvHzbpDjaCedtnobUURjKeiwNR8cM5Tt8yeH4fOh63vbYZWUBv
3jQ5URJX2vHgkOP6sNB/E7/yqYPLxhj3E7Ty2d4/ru3yjJfRiVE=
-----END CERTIFICATE-----

Certificate digest:
66 52 59 16 1F 72 60 BA 9B 48 C2 D2 84 16 86 98
7D 46 9D 8E 4F E0 F8 0E 36 CB 22 72 32 42 80 2C
```

X002

```
Modulus:
A6 17 E0 E4 6B FF E3 0A EC 1A D3 B6 9A 8D F7 53
17 22 0B F0 D2 9B 0D 11 75 B0 61 0B 9B D5 3E CA
A0 9A F9 43 81 9D 5B A6 C1 96 C2 C5 CE 88 15 39
07 1A 5B 08 DB ED 04 B4 BB 8D 1F F8 B5 28 0D 88
3A D6 98 93 CB 33 4E 72 80 52 B3 87 28 7F F0 D7
76 AA D6 42 B7 F2 5A 42 E9 EC 49 F3 69 20 B9 BE
6F D8 75 66 2B 90 46 70 8E B2 FB 09 AB B7 94 0A
BA 89 9D 4E 4B 7F F0 16 58 4E 60 F4 F9 5A 39 C5
34 D4 D4 6A 25 A3 11 C0 E2 38 23 EE 6E D3 52 F5
25 D2 7D EB CA D7 02 48 60 94 82 D2 C3 0D 32 B8
A7 68 E6 A4 FF 0A 43 8F 03 AB B8 7B 88 B3 3F 1B
C7 3C 62 79 E9 D7 7A D6 B4 5F 3C AB DD AE F4 AE
BB A8 04 3B 3A 25 4B A3 3A B9 EA C6 0F AA C8 32
29 0C 86 E4 1F ED 1D 42 1B 97 0B 0A 31 90 28 30
C3 A7 63 5A A9 EC D6 03 5A C3 DF 44 71 6C 86 0B
DB 7A BB 54 B3 95 D2 7E BB DE F8 E2 0E 56 AB 5D


Public exponent:
01 00 01


SHA-256 digest:
26 37 2E F7 BA A0 E7 DE 4F 3E 7F B4 3A FA 81 38
```

```
12 2C 90 C2 22 4A D3 58 61 7C A3 81 61 DE 06 11
```

```
Private exponent:
2F 84 33 4D 85 EC 2D CB 09 22 DF A0 A4 F4 AA 65
F5 FF 42 85 41 EB 23 C0 F3 F7 62 BC 0C 77 E4 3F
D9 D4 9C 2D 08 DE B8 C2 AB 2D 73 49 5D BD A6 BC
AA E0 8A 5C AD 76 50 5B 58 30 96 8D F3 5D A3 09
6F 33 C7 70 B8 B8 53 E0 04 00 24 E6 2C DA F5 4E
DE DF 1E EE F5 6A FB 11 9B 94 82 CF 85 2A DC 98
B3 AC AC 61 6F 63 16 13 CB 6F 3E CC 54 78 EF 7C
66 43 4A 62 84 CC F4 25 A0 5B C3 F7 31 82 F3 75
95 FE 7B 3E 51 89 F0 55 35 D1 BD 27 BA F2 27 4C
86 EC B3 45 D9 7F 4F FF C5 88 03 D5 AF 4A 2A E8
12 00 27 E6 1F 05 C1 97 E0 D0 3D 5A 17 27 8A 3E
98 FE A1 F2 10 BD E4 3C 91 A5 EA EF 0E F4 DC 32
54 83 C8 5F 9D 3F D3 AD 1D 25 A6 D5 F6 0D BC 47
70 51 9A 57 65 28 04 58 66 D3 AD A3 F3 CC A7 47
81 0F 56 BF E0 5F C0 93 5F 08 5F E5 D4 50 5E FD
E8 02 27 0A D4 3F 98 0E 6F 33 68 32 AC B2 57 6D
```

```
Certificate:
-----BEGIN CERTIFICATE-----
MIIDQDCCAiigAwIBAgIBAzANBgkqhkiG9w0BAQsFADBOMQswCQYDVQQGEwJGUjEOMAwGA1UEBxMF
UGFyaXMxEzARBgNVBAoTCkUuQi5JLkMuUy4xCTAHBgNVBAsTADEPMA0GA1UEAxMGRS4gQml4MB4X
DTA5MDcyNzA5MDYyMFoXDTE0MDcyNzA5MDYyMFowTjELMAkGA1UEBhMCRlIxDjAMBgNVBAcTBVBh
cmlzMRMwEQYDVQQKEwpFLkIuSS5DLlMuMQkwBwYDVQQLEwAxDzANBgNVBAMTBkUuIEJpeDCCASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKYX4ORr/+MK7BrTtpqN91MXIgvw0psNEXWwYQub
1T7KoJr5Q4GdW6bBlsLFzogVOQcaWwjb7QS0u40f+LUoDYg61piTyzNOcoBSs4cof/DXdqrWQrfy
WkLp7EnzaSC5vm/YdWYrkEZwjrL7Cau3lAq6iZ1OS3/wFlhOYPT5WjnFNNTUaiWjEcDiOCPubtNS
9SXSfevK1wJIYJSC0sMNMrinaOak/wpDjwOruHuIsz8bxzxieenXeta0Xzyr3a70rruoBDs6JUuj
Ornqxg+qyDIpDIbkH+0dQhuXCwoxkCgww6djWqns1gNaw99EcWyGC9t6u1SzldJ+u9744g5Wq10C
AwEAAaMpMCcwCwYDVR0PBAQDAgeAMAoGA1UdDgQDBAEzMAwGA1UdIwQFMAOAATMwDQYJKoZIhvcN
AQELBQADggEBAAUguaYDREZNnKT4BOoTL7wU+Jh0nG+2Q0tnyMf1JHENbkMGYZsJjsGx9wUu8A6b
mrUxxWCak7jbwGpzSIfjV5Bs/MoOyoJuQM6edQd/aB/JNsDCUk6FcRsaPdf0oPUgP0iupV3aDuh0
Md5yCaYlRsuNMNVllRAzwfWvGKrOw95GEdkt2Nf5/mXk4E7/apmkjZ3o8eyfaf3xzFZiG/ijN36Y
rZDArrvfy4Ir8ppiUax9mUl3Efm2IL9o7d8amijZP3drjF4JT6L0JA1ACZxlO732zdqxWVzHilJl
QUGA7ZACE+PdZ+f0pOrPzIL3MiYvHQUTE96a6vP2IJWj5AFTDIg=
-----END CERTIFICATE-----
```

```
Certificate digest:
7F 9B 13 D5 06 76 C7 23 12 C9 B6 92 0D AE A7 B4
23 C1 3F F8 68 CA 6D C6 DB C8 26 BD B4 5C BE A5
```

E002

```
Modulus:
D6 B9 23 64 5E 93 E0 2E BA F7 80 A6 88 E6 E9 C0
8F 7C 93 89 E0 6A A0 DE 06 92 10 57 C9 A7 DB F4
44 02 B1 23 9E 0C 1A 9C 92 5B 74 DE 75 A4 2C BF
4A 7D 74 66 2B 6A 3E CB 38 98 FA 72 91 C4 C9 C8
D0 F1 3B 37 59 93 5D B0 77 32 28 A1 38 B2 C8 C2
68 23 C3 7E B1 35 81 5A 51 AF 4F C1 11 D9 91 07
39 B6 49 00 65 AA 07 AB 0E 46 34 C4 44 FA AA B1
0F 3E B1 43 5C 4A 84 97 7D C9 CC 56 B6 E0 A5 B5
52 FC BB 20 43 09 67 AB 6A 66 B1 1A B7 E4 39 15
7F ED 37 52 16 2C DD 09 29 B7 38 5A 41 71 85 99
B0 C3 6F 8A 29 4F 1A 75 66 02 6A CE 5E 87 F8 60
C3 84 50 39 FB 1C F5 0B 93 DA E0 E2 F6 8C D3 82
CA E4 6E EB 4C 32 98 7D 60 73 A7 0E 99 7B D8 4C
AC 82 49 56 3E 93 CF F0 3D DA BA AC 5D 72 95 A7
13 11 61 61 6E 2F B5 64 0F 11 54 93 EA 43 33 9E
EA 05 F1 31 71 45 A0 A4 DE 9F 54 ED 92 05 E6 E9
```

```
Public exponent:
01 00 01
```

```
SHA-256 digest:
AF 02 31 49 42 40 54 41 43 43 4B 1D 61 EF 82 C9
B7 6C A4 C9 65 0C 6E 80 AC DB 5D 0B 35 1B 4F 51


Private exponent:
96 FB 43 59 95 EF 20 D9 7A FA 01 6A 18 25 56 03
E2 60 D4 55 44 89 75 67 E8 F2 D7 AF 02 CA 97 2C
E0 8D 8C 04 E1 62 DD 6B 6E B6 04 2D 50 47 0A 77
CF 66 FB 6B C3 E7 47 14 1A 4D E6 FC 9C 66 E5 03
E1 77 5E 0E 03 5C 4D AA 81 85 B1 6F FB B0 2A DE
17 DF DD DC BD BA 43 A4 40 7F A6 F2 B9 1C 64 8F
D2 12 CC AE 0C CE 4B EA 09 75 70 30 F9 D7 D0 22
03 F8 2F 28 90 42 8D 8D 00 13 DF 2C B2 16 B0 2C
CF DF 69 02 83 C3 EF 17 C6 C9 90 DB A5 2A 5D CE
69 BA DB 3E BD CE 70 99 8D E8 F5 42 B9 27 54 D9
A2 DB BF FC 6E 56 AD FB 01 C8 CD 30 62 CA 26 AF
8E 17 02 6D 9D 98 40 58 21 13 F2 DB D4 1F 1D 08
9B E3 ED E8 A1 C1 38 8D 59 D7 3A D6 EF 54 DF 30
8D FC 0C B1 57 8A E5 F7 BF CD 1B F6 D3 E6 37 8A
7F 09 1D DF 20 85 48 C2 C1 4E 2D 3E DC 6C 82 BF
D8 0D EE 4A 24 05 D1 44 6E 47 75 A3 29 78 DD 21


Certificate:
-----BEGIN CERTIFICATE-----
MIIDQDCCAiigAwIBAgIBAjANBgkqhkiG9w0BAQsFADBOMQswCQYDVQQGEwJGUjEOMAwGA1UEBxMF
UGFyaXMxEzARBgNVBAoTCkUuQi5JLkMuUy4xCTAHBgNVBAsTADEPMA0GA1UEAxMGRS4gQml4MB4X
DTA5MDcyNzA5MDYxOVoXDTE0MDcyNzA5MDYxOVowTjELMAkGA1UEBhMCRlIxDjAMBgNVBAcTBVBh
cmlzMRMwEQYDVQQKEwpFLkIuSS5DLlMuMQkwBwYDVQQLEwAxDzANBgNVBAMTBkUuIEJpeDCCASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANa5I2Rek+AuuveApojm6cCPfJOJ4Gqg3gaSEFfJ
p9v0RAKxI54MGpySW3TedaQsv0p9dGYraj7LOJj6cpHEycjQ8Ts3WZNdsHcyKKE4ssjCaCPDfrE1
gVpRr0/BEdmRBzm2SQBlqgerDkY0xET6qrEPPrFDXEqEl33JzFa24KW1Uvy7IEMJZ6tqZrEat+Q5
FX/tN1IWLN0JKbc4WkFxhZmww2+KKU8adWYCas5eh/hgw4RQOfsc9QuT2uDi9ozTgsrkbutMMph9
YHOnDpl72EysgklWPpPP8D3auqxdcpWnExFhYW4vtWQPEVST6kMznuoF8TFxRaCk3p9U7ZIF5ukC
AwEAAaMpMCcwCwYDVR0PBAQDAgUgMAoGA1UdDgQDBAEyMAwGA1UdIwQFMAOAATIwDQYJKoZIhvcN
AQELBQADggEBAFXsZUGzMU2xxjw2W5CkrOnmEx+rz3yPEDU6sdxUmEkd0mehNb2am0IFfLzWdJPU
CS00et32da+skjfn3yFD0/bV0FL8V7QoUK/vkFm7cnPKpMwW/VpDlAVR0PJL4S4R+OFANCyoXdDI
s4hJrSLINeq07CZ8TMZ4Rvn8m2bSShk+lgFyzKyiL9ZDgb+gopWNiOmA73N8ibGEgHuANRFRMfbC
iidpi0lQ9nNTuEP8oOr27sWWl4AWCB9mFS/yAhb0P4pEwoukHGfqzL3BKWNxpc9UxhFtW2CsI3IK
QE7pStVRD7scvx+1Xe89JZmE8nydS6HhAPRRSLmXbvesomHeqJo=
-----END CERTIFICATE-----

Certificate digest:
0A F2 32 2D 11 3F 8C 87 8F B6 19 2B 72 DF E4 2B
13 8B 68 DA 0E 31 65 A8 55 04 8E B9 87 55 96 79
```

Sending a Standard Upload with Keys

First step is to publish the user's keys to the bank via INI und HIA.

INI

Modulus and public exponent of the A005 key are base64 encoded, and the order data for the INI request is created:

```
<?xml version="1.0" encoding="UTF-8"?>
<SignaturePubKeyOrderData xmlns="http://www.ebics.org/S001"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xsi:schemaLocation="http://www.ebics.org/S001
http://www.ebics.org/S001/ebics_signature.xsd">
  <SignaturePubKeyInfo>
    <PubKeyValue>
      <ds:RSAKeyValue>
```

```
<ds:Modulus>ANeOaO2fHl56a7bcS4FAnfTyvGiibmiyed9Jx1wifCojuzzLpnSVWnbDm2wyB1/YXK1V/cl
lK+LCrav48xMnoga0aRcVxrSCtpAW+PB6Wn1hKkNW2m/wIuP1Vg+LB2EAsAVvCiMrCpyGKUUGNQ5x0Ph913
xYUgZ41Rq/CNJ2w4AtuCgbr5KvRTwvKEv4JEJ5qymedq3Yy1kt6t5HbQkLzMpMOBtMLzUBiZUv1sTApEtK3
9CdCI8GAEsZdW5ePSdr77y0qsA7QCTxWliLpNDBx4BOuYJz9IdejqtNT4zAD7sQB6Yai54LzJ0lpHmA3baK
3/Y3gBJqxC3WHSBf68x66ic=</ds:Modulus>
        <ds:Exponent>AQAB</ds:Exponent>
      </ds:RSAKeyValue>
      <TimeStamp>2009-07-28T16:32:48.120+02:00</TimeStamp>
    </PubKeyValue>
    <SignatureVersion>A005</SignatureVersion>
  </SignaturePubKeyInfo>
  <PartnerID>EBICS</PartnerID>
  <UserID>EBIX</UserID>
</SignaturePubKeyOrderData>
```

The order data is then zipped, base64 encoded and put into the INI request:

```
<?xml version="1.0" encoding="UTF-8"?>
<ebicsUnsecuredRequest xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_keymgmt_request.xsd">
  <header authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <PartnerID>EBICS</PartnerID>
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.3, SIZ/PPI</Product>
      <OrderDetails>
        <OrderType>INI</OrderType>
        <OrderID>A025</OrderID>
        <OrderAttribute>DZNNN</OrderAttribute>
      </OrderDetails>
      <SecurityMedium>0000</SecurityMedium>
    </static>
    <mutable/>
  </header>
  <body>
    <DataTransfer>

<OrderData>eNp1k1lzokAUhd+nav6D5Txa2uwuhaYAnajgCmrMy1QDLXYCDdKs/vqhMMtkUj72Od+FvrfP
lR+KwG9kKKY4JMMm22GaDUSc0MXEGzZ31u92r/kw+vlDNrFHYJLGaJ3aOipXsYviMUxgo6ondNg8J0k0ACD
P8w6ysUM7YewBk2HY5o0YFBR/oXK+RrgKAU8Lw3TOKIBtTGgCiYPeq1x6p4gBTB9UjEux96uiKR7Q+hNG6M
CkbubulRp3HVAf/9D3ZjsFdZtV+43G/wOYkVNYG5V1U/bQT9GbVIkuHWxN5Zt+cxahm/opHSlLtIIr7jT1R
Ql2bccUfivkZJXZI8Z2gEvk9ucFm+OTFr6k16sREXN/IPY44PJSZcHxSWf3wPF9vWVoMcyEXrEgoQcZuHX2
RWxqSaQcWmtVOhD2rL8uD1wA8lm6Zvdey1C5iUKVfabhRaxF5aO+2z0uN2LBrM99li+OO+9ZYLcXoC3nXC4
oSap5dizq2dbKM32SCfPJXLyUAXIv/LFkXxMpEaf25tW4LqLFSk0WxnWn4uddxlJLiSaJzvc1V5v1HpUJfX
YPIlqbbtztlsyFKt2NZhUHHxvRcqwWgrpKj/Nrf+ail0uytISrMu7SjSodIRYF4zpn/GgaKLwNdR4ceU+dX
wqNP0xN9ST1CknCzlAG/wz56+gnRRQSRJKRslHUmvtQPt4O3Hk82cIBMhMYRKMqgv02021zPYuVBjw3EHod
lmNaDDdgGBl8gm8ZAd9D8hmp/W3/RgrDiDL4Jtf5A/cCKK9hnBAUz8ajiTrTzOpXH0Lt7+i7+SSDt0O10eD
eSo/+AvzYTgo=</OrderData>
    </DataTransfer>
  </body>
</ebicsUnsecuredRequest>
```

The INI request is then send to the server, resulting in the following reponse:

```
<?xml version="1.0" encoding="UTF-8"?>
<ebicsKeyManagementResponse xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_keymgmt_response.xsd">
  <header authenticate="true">
    <static/>
    <mutable>
      <ReturnCode>000000</ReturnCode>
      <ReportText>[EBICS_OK] OK</ReportText>
```

```
    </mutable>
  </header>
  <body>
    <ReturnCode authenticate="true">000000</ReturnCode>
    <TimestampBankParameter authenticate="true">2009-06-
16T17:56:05.698+02:00</TimestampBankParameter>
  </body>
</ebicsKeyManagementResponse>
```

HIA

The same is done for the X002 and the E002 key, resulting in the following order data:

```
<?xml version="1.0" encoding="UTF-8"?>
<HIARequestOrderData xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_orders.xsd">
  <AuthenticationPubKeyInfo>
    <PubKeyValue>
      <ds:RSAKeyValue>

<ds:Modulus>AKYX4ORr/+MK7BrTtpqN91MXIgvw0psNEXWwYQub1T7KoJr5Q4GdW6bBlsLFzogVOQcaWwj
b7QS0u40f+LUoDYg61piTyzNOcoBSs4cof/DXdqrWQrfyWkLp7EnzaSC5vm/YdWYrkEZwjrL7Cau3lAq6iZ
1OS3/wFlhOYPT5WjnFNNTUaiWjEcDiOCPubtNS9SXSfevK1wJIYJSC0sMNMrinaOak/wpDjwOruHuIsz8bx
zxieenXeta0Xzyr3a70rruoBDs6JUujOrnqxg+qyDIpDIbkH+0dQhuXCwoxkCgww6djWqns1gNaw99EcWyG
C9t6u1SzldJ+u9744g5Wq10=</ds:Modulus>
        <ds:Exponent>AQAB</ds:Exponent>
      </ds:RSAKeyValue>
    </PubKeyValue>
    <AuthenticationVersion>X002</AuthenticationVersion>
  </AuthenticationPubKeyInfo>
  <EncryptionPubKeyInfo>
    <PubKeyValue>
      <ds:RSAKeyValue>

<ds:Modulus>ANa5I2Rek+AuuveApojm6cCPfJOJ4Gqg3gaSEFfJp9v0RAKxI54MGpySW3TedaQsv0p9dGY
raj7LOJj6cpHEycjQ8Ts3WZNdsHcyKKE4ssjCaCPDfrE1gVpRr0/BEdmRBzm2SQBlqgerDkY0xET6qrEPPr
FDXEqEl33JzFa24KW1Uvy7IEMJZ6tqZrEat+Q5FX/tN1IWLN0JKbc4WkFxhZmww2+KKU8adWYCas5eh/hgw
4RQOfsc9QuT2uDi9ozTgsrkbutMMph9YHOnDpl72EysgklWPpPP8D3auqxdcpWnExFhYW4vtWQPEVST6kMz
nuoF8TFxRaCk3p9U7ZIF5uk=</ds:Modulus>
        <ds:Exponent>AQAB</ds:Exponent>
      </ds:RSAKeyValue>
    </PubKeyValue>
    <EncryptionVersion>E002</EncryptionVersion>
  </EncryptionPubKeyInfo>
  <PartnerID>EBICS</PartnerID>
  <UserID>EBIX</UserID>
</HIARequestOrderData>
```

and the following HIA request:

```
<?xml version="1.0" encoding="UTF-8"?>
<ebicsUnsecuredRequest xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_keymgmt_request.xsd">
  <header authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <PartnerID>EBICS</PartnerID>
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.3, SIZ/PPI</Product>
      <OrderDetails>
        <OrderType>HIA</OrderType>
        <OrderID>A026</OrderID>
        <OrderAttribute>DZNNN</OrderAttribute>
```

```
      </OrderDetails>
      <SecurityMedium>0000</SecurityMedium>
    </static>
    <mutable/>
  </header>
  <body>
    <DataTransfer>

<OrderData>eNq9lU+TqkYUxfepynewzNJ6AwiiTDm+UmhGQEEEBd2kWkAElD/dtICfPkTfTPIyM1mlsuxz
zm24fX/Q4+/15dy5BghHWfrSZZ7obidIvcyP0vClu7Hlb6Pu98mvv4znynQdFCTApYH8AEmwhJ22NMUv3VN
Z5s8UVVXVU3CIPPyUoZCa0zTbfSSeaxz9lKrYe6RP0wzlLheWdwou8FuU4hKmXvBW5eMvimiKFqg24+Mo/K
1N4+gZ37dYZB4s7318+UqdLx3qvvw9+7M7/FRjv9u23emMp6Q8BWkZPbZekYMWNEp6zO5u6z+ULTyT4IfUi
j5+XlvTD/rDWWY+ORM8mWo7lzPWiOotteEM2WVe6AKzdJXwWtE51oHrVDuTHBh7qGUqGpjcq+/wh9kZL+Rb
Fm4N04NOFR+GpkUTjj72FptM2oU8k0d2c9MNL5tZmPOyIyW5foEcEx0bJ1nkQ5DeoCUOrhdq5zs7lIB9FaP
FUISEPU8LPtozhsVSlXw+GbuVPXDiVNZ1ewMjJwaeFBli23OpW4LlWsfgqjGVquxUS6TxUl+iKIUGTKgql+
LKQGROFHwbHepbHQVB6gYlpN1bg1g4pBEi2UzCvLohsYHSog57RSMpuaQcknmP9s0TccUqqxMxrCrej50ix
Uyow0oQgOc0r6JQ8oSxbmdf7RFhyHHhwCkY+mVM/e2Qfz56UOdZ2k5zMjWns3vuXXmfHfXp8MbUx0H/g43t
4yuauDTdH1Ofe3ekqH9lagxSDzX5f06bDgdKfx0kvSkh12CaZ/GF98TVUTVU7rUI2RBaQD6quXCl11OtVgb
c8jVvLIe1Ax+a+Erngv+6QzAeLgw15r18DhovNkc2Zp297uO512ga4DCORSiupCMCTLjN14imZsC/rGe3S9
8yZ+cibP8eyY6ugc0XCKxWSJZcUIAzy6o3GfY5zWE212aogKW658tijwAse+ZAdqlSZxRnodOqdvA4J5Hr0
/5SVf2epm1GsCVZhHgQnKhTWHFr0zhiTzCJ3SdSJGQ3O8QoOZByucxPwm5upFJ+HvZBg8Pk7Kzy1WoksZAU
te/lTgpq+bRzuGvpmCuwtWw+Wd5SkskjW67XUEzYXNgM94o8IMn/SNtfXLzRBO6kfdQflH3J0XgFUZkGSJE
mYKaIVvu0d+Hub/Cb6Y6pH4v2DqA+uQQmfwBrfAqU</OrderData>
    </DataTransfer>
  </body>
</ebicsUnsecuredRequest>
```

The response is identical to the INI response.

Confirmation of the keys

The keys (modulus, public exponent, hash value) are printed, signed and sent to the bank. The bank verifies the keys by, e.g. by comparing the printed hash values with those it calculated itself from the transmitted keys. After the confirmation, the user may (and must) use these keys.

HPB

Usually, the first order a user performs after the confirmation of his keys is the download of the bank's key. For this order, the encryption is already in effect, as this is done with the user's encryption key. The request is also X00 signed by the user. However, the bank's response cannot be signed, as the user doesn't yet know the bank's public key.
For the download request, first a random nonce is chosen:

```
C2 E0 19 E2 68 96 7E CA 5F F2 2D F3 89 18 6D B0
```

A request is compiled for the given host id, partner id and user id.

```
<ebicsNoPubKeyDigestsRequest Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_keymgmt_request.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <header authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <Nonce>C2E019E268967ECA5FF22DF389186DB0</Nonce>
      <Timestamp>2009-08-03T12:11:41.385Z</Timestamp>
      <PartnerID>EBICS</PartnerID>
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.4, SIZ/PPI</Product>
      <OrderDetails>
```

```
        <OrderType>HPB</OrderType>
        <OrderAttribute>DZHNN</OrderAttribute>
      </OrderDetails>
      <SecurityMedium>0000</SecurityMedium>
    </static>
    <mutable/>
  </header>
  <body/>
</ebicsNoPubKeyDigestsRequest>
```

All the Elements with the attribute authenticate = true and their sub elements are considered for the message digest calculation:

```
<header xmlns="http://www.ebics.org/H003"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <Nonce>C2E019E268967ECA5FF22DF389186DB0</Nonce>
      <Timestamp>2009-08-03T12:11:41.385Z</Timestamp>
      <PartnerID>EBICS</PartnerID>
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.4, SIZ/PPI</Product>
      <OrderDetails>
        <OrderType>HPB</OrderType>
        <OrderAttribute>DZHNN</OrderAttribute>
      </OrderDetails>
      <SecurityMedium>0000</SecurityMedium>
    </static>
    <mutable></mutable>
  </header>
```

The digest for this XML fragment is:

```
C1 7A A3 E4 93 08 87 F6 4B D7 F5 E6 DB 06 5E 71
0C 3B E2 52 F9 17 2E 26 A9 CE 66 C3 5D F9 82 FA
```

This message digest is put into a SignedInfo XML fragment, the namespace declaration already as it will appear in the final document:

```
<ds:SignedInfo xmlns="http://www.ebics.org/H003"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-
xml-c14n-20010315"></ds:CanonicalizationMethod>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"></ds:SignatureMethod>
      <ds:Reference URI="#xpointer(//*[@authenticate='true'])">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"></ds:Transform>
        </ds:Transforms>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"></ds:DigestMethod>

<ds:DigestValue>wXqj5JMIh/ZL1/Xm2wZecQw74lL5Fy4mqc5mw135gvo=</ds:DigestValu
e>
      </ds:Reference>
    </ds:SignedInfo>
```

This SignedInfo fragment is then actually signed. For this, first it's message digest is calculated which is:

```
D0 A3 3E 51 81 E0 EF 86 D0 23 86 3D 39 D3 1B C9
2E ED 75 14 70 F5 FD 97 C9 34 1F DD 81 C7 3C E8
```

This digest is then completed with an ASN.1 prefix:

```
30 31 30 0D 06 09 60 86 48 01 65 03 04 02 01 05
00 04 20 D0 A3 3E 51 81 E0 EF 86 D0 23 86 3D 39
D3 1B C9 2E ED 75 14 70 F5 FD 97 C9 34 1F DD 81
C7 3C E8
```

This is signed with the bank user's private key:

```
00 3F 24 5C 6D 9B 26 AD 08 FC B5 CA 89 17 AF 31
32 3F 2D A0 F8 C5 89 8E 5F 78 46 5E 1A 19 01 A3
8E F1 2C AD 09 34 63 F5 F3 E0 2A 6D 12 F3 A8 5A
1D 14 C2 44 B6 98 C1 83 24 34 75 B8 44 92 FA 07
62 5B 31 D9 48 B7 F8 CE 3C AD D1 FE 36 5A 51 5E
94 5D BA AE 59 FD 75 25 AB FA 29 6E 02 37 0E B5
54 A5 4E 10 E7 47 9D 52 DC A4 04 16 34 62 73 39
93 DF 25 14 85 7E 22 53 0B D6 3A 00 D3 2F 1A 37
D5 F6 F6 63 4D 64 F4 41 E7 2F 10 F0 FA F0 D5 FE
5D 00 4D 26 C2 11 9A F0 0C 3E 54 75 BD 2D C1 63
9F 19 45 44 55 07 5B 95 74 2F 68 DF 51 DB EF 7C
84 0E 89 0A 61 7E FE 5F FE 9A 17 0E 94 E8 88 59
1B 0A B1 75 2A A7 AB CB D0 94 29 4B FB 7D 72 A9
8A AF FD 57 A3 A1 A2 BE 54 C4 47 96 74 ED 3B 72
BD 1D 45 A9 85 29 46 08 B1 28 2C 46 43 82 4D 0E
F1 58 65 D7 8E FF D4 FE 34 E8 5C F6 B4 B3 BA F7
```

The complete signature is then base64 encoded and added to the original message:

```
<ebicsNoPubKeyDigestsRequest Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_keymgmt_request.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <header authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <Nonce>C2E019E268967ECA5FF22DF389186DB0</Nonce>
      <Timestamp>2009-08-03T12:11:41.385Z</Timestamp>
      <PartnerID>EBICS</PartnerID>
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.4, SIZ/PPI</Product>
      <OrderDetails>
        <OrderType>HPB</OrderType>
        <OrderAttribute>DZHNN</OrderAttribute>
      </OrderDetails>
      <SecurityMedium>0000</SecurityMedium>
    </static>
    <mutable/>
  </header>
  <AuthSignature>
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-
xml-c14n-20010315"/>
```

```
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"/>
      <ds:Reference URI="#xpointer(//*[@authenticate='true'])">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"/>
        </ds:Transforms>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>wXqj5JMIh/ZL1/Xm2wZecQw74lL5Fy4mqc5mw135gvo=</ds:DigestValu
e>
      </ds:Reference>
    </ds:SignedInfo>

<ds:SignatureValue>AD8kXG2bJq0I/LXKiRevMTI/LaD4xYmOX3hGXhoZAaOO8SytCTRj9fPg
Km0S86haHRTCRLaYwYMkNHW4RJL6B2JbMdlIt/jOPK3R/jZaUV6UXbquWf11Jav6KW4CNw61VKV
OEOdHnVLcpAQWNGJzOZPfJRSFfiJTC9Y6ANMvGjfV9vZjTWT0QecvEPD68NX+XQBNJsIRmvAMPl
R1vS3BY58ZRURVB1uVdC9o31Hb73yEDokKYX7+X/6aFw6U6IhZGwqxdSqnq8vQlClL+31yqYqv/
VejoaK+VMRHlnTtO3K9HUWphSlGCLEoLEZDgk0O8Vhl147/1P406Fz2tLO69w==</ds:Signatu
reValue>
  </AuthSignature>
  <body/>
</ebicsNoPubKeyDigestsRequest>
```

This message is then sent to the EBICS server. From there, we get the following response:

```
<ebicsKeyManagementResponse Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_keymgmt_response.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <header authenticate="true">
    <static/>
    <mutable>
      <ReturnCode>000000</ReturnCode>
      <ReportText>[EBICS_OK] OK</ReportText>
    </mutable>
  </header>
  <body>
    <DataTransfer>
      <DataEncryptionInfo authenticate="true">
        <EncryptionPubKeyDigest
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="E002">rwIxSUJAVEFDQ0sdYe+CybdspMllDG6ArNtdCzUbT1E=</EncryptionPubK
eyDigest>

<TransactionKey>CPdd6ODXrCxHWpYrwS/5a3NIZ1R2SDscHHJSONc2I7g8ara7qv5Ra7H03W+
336DPQLZQxH+EBRWdE2FQ1vrVTcypK6fkFarmO1dEtQt6Ka89AozC7aQMyp4yGl+zlF+XvFkJXZ
bRXgoEOPsgcG0xM6nKLXnr1ThYRFOoAasigLDGwqxliyhoR94F9dL7YB8Le/uZeRj78mNp4smkE
5ZKKyPhwVx+0IW78YuVCSC2c2qbsMmm4evAvycdpLfuuoG4KVlNMIlvkKeBfMZFELcmeKKm85D2
zPYkZ7cqG2IfAAp5qdS6K8P7RY4gtjUhOtP5mi8bZt/ZcjioUmIR9dAMnA==</TransactionKe
y>
      </DataEncryptionInfo>

<OrderData>lJ/v1HvOYAZKTPkxULH4mOcUy01+EaAsxLzCPhtieAo9m5AtU5URdyBVfmclHbtW
e3ELb02lR37bChynq0Esqpljxx7CiB+kl/bVs7XkNQQmtGPlh08O7ApJ7AHdKPFFvE+SM2OWpSw
UrqJGtmyaU6s+RM66o/3Q0wvKi4y3+Q2jm8nkiWZpPfNt52g4ReE9Ge86o+L+EBOBgZm634L5Z/
i6pq908mXBAklPdk1ERdEgWONaOeYYmB70FToplR1gINJEF2CnfAtD/wV+DuQFPDth+EWHqhe3a
4xO0d9eJ55ou2QCYOPUC2g/qz1sp7GR74W7P5IWJdmnUEuPFsKJ6OvM+1V8S06jZGDa/QhUGDK8
F3G/TMP09W1MsrlQTRr3YQ5H43+pMY7I6Jysze+X9OKZZTTpv4GmYm1USjmraGyMuPF3j9hrlmp
ttK+1gg2WzytCJO44YsZp9ckls8JNK4bT5uwTuomKZoRgmiIOTcMh4VP7V28TWj2XdF+T6rBFIL
```

```
TvQyXMOzvbe1kLwg2O31JDBdJ8X1fHCYJxKsIrYCSDPz4sha8h7QmHrCSFmjoXb+LUElnbLXZVR
kY5/Dc73EM6Es3wAgjwSJK2asRqdqU0PhbS5qpIlgYKsNcb+Wes</OrderData>
    </DataTransfer>
    <ReturnCode authenticate="true">000000</ReturnCode>
    <TimestampBankParameter authenticate="true">2009-06-
16T17:56:05.698+02:00</TimestampBankParameter>
  </body>
</ebicsKeyManagementResponse>
```

This response contains the transaction key randomly chosen by the bank, and the first (and in this case only) segment of download data.
The transaction key's base64 encoding is removed:

```
08 F7 5D E8 E0 D7 AC 2C 47 5A 96 2B C1 2F F9 6B
73 48 67 54 76 48 3B 1C 1C 72 52 38 D7 36 23 B8
3C 6A B6 BB AA FE 51 6B B1 F4 DD 6F B7 DF A0 CF
40 B6 50 C4 7F 84 05 15 9D 13 61 50 D6 FA D5 4D
CC A9 2B A7 E4 15 AA E6 3B 57 44 B5 0B 7A 29 AF
3D 02 8C C2 ED A4 0C CA 9E 32 1A 5F B3 94 5F 97
BC 59 09 5D 96 D1 5E 0A 04 38 FB 20 70 6D 31 33
A9 CA 2D 79 EB D5 38 58 44 53 A8 01 AB 22 80 B0
C6 C2 AC 65 8B 28 68 47 DE 05 F5 D2 FB 60 1F 0B
7B FB 99 79 18 FB F2 63 69 E2 C9 A4 13 96 4A 2B
23 E1 C1 5C 7E D0 85 BB F1 8B 95 09 20 B6 73 6A
9B B0 C9 A6 E1 EB C0 BF 27 1D A4 B7 EE BA 81 B8
29 59 4D 30 89 6F 90 A7 81 7C C6 45 10 B7 26 78
A2 A6 F3 90 F6 CC F6 24 67 B7 2A 1B 62 1F 00 0A
79 A9 D4 BA 2B C3 FB 45 8E 20 B6 35 21 3A D3 F9
9A 2F 1B 66 DF D9 72 38 A8 52 62 11 F5 D0 0C 9C
```

Decrypting this with the user's private encryption key results in:

```
99 DE DF F3 FE 42 CA 55 42 34 88 93 3B DF A7 79
```

which is the AES key for this transaction. So when the order data is base64 decoded, it can be decrypted. The result is a zip stream, which after decompression yields the following order data:

```
<xml-fragment xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_orders.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:h003="http://www.ebics.org/H003">
  <h003:AuthenticationPubKeyInfo>
    <h003:PubKeyValue>
      <ds:RSAKeyValue>

<ds:Modulus>AJ2/0OoIZydb9sgKbiwqDcwA0NtcUMIYi6GK7PqoRszu1uytGnxJjQhGt62kMsl
WvLgebUSwdq/T3YyGsa3KQeIGaUUn9iqyu3BoNOMdo2DLN4NdGMY1WR/HbYRKR3JHyURhTBKw27
KScRRdKGl6jY+VcXcTKJL8PjXMQH5cD6Gz</ds:Modulus>
        <ds:Exponent>AQAB</ds:Exponent>
      </ds:RSAKeyValue>
    </h003:PubKeyValue>
    <h003:AuthenticationVersion>X002</h003:AuthenticationVersion>
  </h003:AuthenticationPubKeyInfo>
  <h003:EncryptionPubKeyInfo>
    <h003:PubKeyValue>
      <ds:RSAKeyValue>

<ds:Modulus>AJ2/0OoIZydb9sgKbiwqDcwA0NtcUMIYi6GK7PqoRszu1uytGnxJjQhGt62kMsl
```

```
WvLgebUSwdq/T3YyGsa3KQeIGaUUn9iqyu3BoNOMdo2DLN4NdGMY1WR/HbYRKR3JHyURhTBKw27
KScRRdKGl6jY+VcXcTKJL8PjXMQH5cD6Gz</ds:Modulus>
        <ds:Exponent>AQAB</ds:Exponent>
      </ds:RSAKeyValue>
    </h003:PubKeyValue>
    <h003:EncryptionVersion>E002</h003:EncryptionVersion>
  </h003:EncryptionPubKeyInfo>
  <h003:HostID>SIZBN001</h003:HostID>
</xml-fragment>
```


The contained key values are just the base64 representation of the bank keys as they are printed above.

Upload

After downloading and verifying the bank’s keys, a first upload may be performed. In this example we will upload some binary data (the EBICS logo as provided with this document) using the order type “FTB”. But before we can send it, we have to create a signature.

Signature Creation

The SHA-256 digest of the order data are calculated:

```
98 C6 69 83 C7 E0 B8 29 5F B1 25 3A E0 25 B8 2C
A7 69 EE 68 46 34 3D CF D7 1F 17 7A 60 60 E7 26
```

This digest is padded, resulting in the following DSI:

```
01 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF 00 30 31 30 0D
06 09 60 86 48 01 65 03 04 02 01 05 00 04 20 98
C6 69 83 C7 E0 B8 29 5F B1 25 3A E0 25 B8 2C A7
69 EE 68 46 34 3D CF D7 1F 17 7A 60 60 E7 26
```

The DSI is signed with the users A005 key:

```
11 93 D5 9B A3 BF 57 8F 8C 08 4F F7 05 3D 8A 5B
A0 90 85 34 28 21 3F CF BC F1 39 B3 E1 BD D7 10
C5 48 31 2F 5C 1E 99 63 50 3C E5 40 C3 13 3C EC
04 6A 3D B7 2F F9 46 EA DF 1B 71 FA 76 A3 55 93
FA 19 77 CF 5A 5B CB 71 F0 BF 53 C1 A7 58 E0 09
35 AD 61 AB 40 D1 E2 6E 55 48 46 6E ED CA 8D DD
A4 3E 35 97 73 D8 67 45 5B 3A 2D 84 FA 75 54 08
98 8A A0 5A 9F E8 EC 71 6E 0F 08 66 38 62 46 7C
EA 98 CB D1 7E 2E 3C 30 32 AE E3 18 DA 78 30 3A
68 CB D9 E2 E7 F8 A0 BF AE D1 1D 98 2A F5 96 9B
55 51 1C AD 62 1C E9 B7 CD 16 AE FE 53 8F 16 82
03 16 4D 83 DA 36 B1 32 FA A1 20 CE 93 93 34 7A
DE 6A 4F 32 19 82 74 A3 85 1C D2 B2 C0 D2 8E A2
```

```
CA E1 FF E6 AE 6E 89 2D 1A 7D 10 70 83 FC 94 69
37 CD 20 52 E9 2B EF 4C 00 E8 B1 4A 8C FA 47 30
76 C1 B8 47 19 D7 D3 1E 2B 93 2D 5B 91 02 97 09
```

The result is base64 encoded and put into an user signature data document:

```
<?xml version="1.0" encoding="UTF-8"?>
<UserSignatureData xmlns="http://www.ebics.org/S001"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ebics.org/S001
http://www.ebics.org/S001/ebics_signature.xsd">
  <OrderSignatureData>
    <SignatureVersion>A005</SignatureVersion>

<SignatureValue>EZPVm6O/V4+MCE/3BT2KW6CQhTQoIT/PvPE5s+G91xDFSDEvXB6ZY1A85UD
DEzzsBGo9ty/5RurfG3H6dqNVk/oZd89aW8tx8L9TwadY4Ak1rWGrQNHiblVIRm7tyo3dpD41l3
PYZ0VbOi2E+nVUCJiKoFqf6Oxxbg8IZjhiRnzqmMvRfi48MDKu4xjaeDA6aMvZ4uf4oL+u0R2YK
vWWm1VRHK1iHOm3zRau/lOPFoIDFk2D2jaxMvqhIM6TkzR63mpPMhmCdKOFHNKywNKOosrh/+au
boktGn0QcIP8lGk3zSBS6SvvTADosUqM+kcwdsG4RxnX0x4rky1bkQKXCQ==</SignatureValu
e>
    <PartnerID>EBICS</PartnerID>
    <UserID>EBIX</UserID>
  </OrderSignatureData>
</UserSignatureData>
```

Initialisation phase

As this is an upload, the client chooses a nonce

```
A5 48 8F 43 22 30 63 17 1C A0 FA 59 AD C6 35 F0
```

and the transaction key:

```
C8 56 F5 FD 58 EB 76 A4 D0 3C 95 AD 78 AF CA 14
```

The transaction key is encrypted using the bank’s public key, resulting in

```
4A 13 27 47 FF CA 78 2F F7 9C A1 B3 37 74 64 61
3F 14 85 44 85 A7 86 17 92 30 D9 27 4F C6 38 4F
05 D7 5E B9 25 D3 7F 1C 90 07 CA E7 39 B6 24 9A
94 8D DA C9 3F 57 D5 76 2D 05 05 B9 5B 45 FF 08
17 97 24 FE 41 24 DC D7 5A E3 C9 58 83 46 7D 0F
4D 2A 75 A7 4F CA 4D B6 F2 34 3E 9B D4 F0 16 E1
FB 1B F8 F8 4B 55 0A 89 25 41 DB AB A8 94 B6 DE
FF F5 56 07 67 04 5F 45 19 05 6C 2D 26 10 BE 43
```

The base64 equivalents of the nonce and the encrypted key in a suitable XML structure are then incorporated into the initial upload request, as well as the digests of the bank’s public keys. The number of segments is calculated and inserted. The order signature we created above is compressed, encrypted with the new transaction key and also put into the request XML.

```
<ebicsRequest Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_request.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <header authenticate="true">
    <static>
```

```
      <HostID>SIZBN001</HostID>
      <Nonce>A5488F43223063171CA0FA59ADC635F0</Nonce>
      <Timestamp>2009-08-04T08:41:56.967Z</Timestamp>
      <PartnerID>EBICS</PartnerID>
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.4, SIZ/PPI</Product>
      <OrderDetails>
        <OrderType>FTB</OrderType>
        <OrderID>A037</OrderID>
        <OrderAttribute>OZHNN</OrderAttribute>
        <StandardOrderParams/>
      </OrderDetails>
      <BankPubKeyDigests>
        <Authentication Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="X002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</Authentication
>
        <Encryption Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="E002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</Encryption>
      </BankPubKeyDigests>
      <SecurityMedium>0000</SecurityMedium>
      <NumSegments>1</NumSegments>
    </static>
    <mutable>
      <TransactionPhase>Initialisation</TransactionPhase>
    </mutable>
  </header>
  <AuthSignature/>
  <body>
    <DataTransfer>
      <DataEncryptionInfo authenticate="true">
        <EncryptionPubKeyDigest
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="E002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</EncryptionPubK
eyDigest>

<TransactionKey>ShMnR//KeC/3nKGzN3RkYT8UhUSFp4YXkjDZJ0/GOE8F1165JdN/HJAHyuc
5tiSalI3ayT9X1XYtBQW5W0X/CBeXJP5BJNzXWuPJWINGfQ9NKnWnT8pNtvI0PpvU8Bbh+xv4+E
tVCoklQdurqJS23v/1VgdnBF9FGQVsLSYQvkM=</TransactionKey>
      </DataEncryptionInfo>
      <SignatureData
authenticate="true">yuSeXqH2iy44fj9tokf9JzPE1y7VnPrBIv3o0enAiQ+oP2nEFjyX5Ch
XZDeuVcoEQB9dlKMxS7YUQ64NOuzkUTKccy7ZF79Ctd9on+fjg9e/utO26gZh2+JPtycOUYlpfN
oUuaA2No5JYMehUR8x43iFpw1vCT3bVqVuCxjbxJVHES6kf9kQMMbAEk96vY1Fzart3yLDVTFfX
gkqHowP42FHPdFKjSKKhkHm2poGD9N9jtCuh+SU+52A02zBTUR/1wADGIzPScDdNE1sm85e+XEt
FhuBKI9E5TRb5HQ9Ne1RG0P9HAWFBwGgls1dbAoD3bMxkZkvdeResJipbIp99NitonCA8bJQ3Is
YwN8RXQSul1FhiFcbqGUW575/QHE9YYoHX9Zjp8lt8xHDyq10cSulWW6T+1x06x63ySARrXQWcM
VyF5ukffJqvb3PECP9CED5W4SZ9mA/GqPTMLgQQmwbn4TZkY5yU0TdUw8edjM6nxyXeZpJ42Jiv
Lm1Bwi/lRYzzyR+P/qrUsYgQQM+tXrdzAXi0X9fZfkmIhTeg2ScmKRhgcHncThw7SvdxD9t9KoL
IrlouR6cltJp6xdGCxCnSs8JjjX6jY0+uzriahYH+t4nakGapwswSyDoq4VkGreLQCeSIBxYzI7
+dZ7UUUZS653otxb/Uho2ReDiwfRjqRB7XU+1vqGAL9N6gEOB</SignatureData>
    </DataTransfer>
  </body>
</ebicsRequest>
```

This request is signed just as the HPB request was. The XML fragment containing the parts with the authenticate attribute set to true looks like this:

```
<header xmlns="http://www.ebics.org/H003"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
```

```
      <Nonce>A5488F43223063171CA0FA59ADC635F0</Nonce>
      <Timestamp>2009-08-04T08:41:56.967Z</Timestamp>
      <PartnerID>EBICS</PartnerID>
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.4, SIZ/PPI</Product>
      <OrderDetails>
        <OrderType>FTB</OrderType>
        <OrderID>A037</OrderID>
        <OrderAttribute>OZHNN</OrderAttribute>
        <StandardOrderParams></StandardOrderParams>
      </OrderDetails>
      <BankPubKeyDigests>
        <Authentication Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="X002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</Authentication
>
        <Encryption Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="E002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</Encryption>
      </BankPubKeyDigests>
      <SecurityMedium>0000</SecurityMedium>
      <NumSegments>1</NumSegments>
    </static>
    <mutable>
      <TransactionPhase>Initialisation</TransactionPhase>
    </mutable>
  </header><DataEncryptionInfo xmlns="http://www.ebics.org/H003"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" authenticate="true">
        <EncryptionPubKeyDigest
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="E002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</EncryptionPubK
eyDigest>

<TransactionKey>ShMnR//KeC/3nKGzN3RkYT8UhUSFp4YXkjDZJ0/GOE8F1165JdN/HJAHyuc
5tiSalI3ayT9X1XYtBQW5W0X/CBeXJP5BJNzXWuPJWINGfQ9NKnWnT8pNtvI0PpvU8Bbh+xv4+E
tVCoklQdurqJS23v/1VgdnBF9FGQVsLSYQvkM=</TransactionKey>
      </DataEncryptionInfo><SignatureData xmlns="http://www.ebics.org/H003"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
authenticate="true">yuSeXqH2iy44fj9tokf9JzPE1y7VnPrBIv3o0enAiQ+oP2nEFjyX5Ch
XZDeuVcoEQB9dlKMxS7YUQ64NOuzkUTKccy7ZF79Ctd9on+fjg9e/utO26gZh2+JPtycOUYlpfN
oUuaA2No5JYMehUR8x43iFpw1vCT3bVqVuCxjbxJVHES6kf9kQMMbAEk96vY1Fzart3yLDVTFfX
gkqHowP42FHPdFKjSKKhkHm2poGD9N9jtCuh+SU+52A02zBTUR/1wADGIzPScDdNE1sm85e+XEt
FhuBKI9E5TRb5HQ9Ne1RG0P9HAWFBwGgls1dbAoD3bMxkZkvdeResJipbIp99NitonCA8bJQ3Is
YwN8RXQSul1FhiFcbqGUW575/QHE9YYoHX9Zjp8lt8xHDyq10cSulWW6T+1x06x63ySARrXQWcM
VyF5ukffJqvb3PECP9CED5W4SZ9mA/GqPTMLgQQmwbn4TZkY5yU0TdUw8edjM6nxyXeZpJ42Jiv
Lm1Bwi/lRYzzyR+P/qrUsYgQQM+tXrdzAXi0X9fZfkmIhTeg2ScmKRhgcHncThw7SvdxD9t9KoL
IrlouR6cltJp6xdGCxCnSs8JjjX6jY0+uzriahYH+t4nakGapwswSyDoq4VkGreLQCeSIBxYzI7
+dZ7UUUZS653otxb/Uho2ReDiwfRjqRB7XU+1vqGAL9N6gEOB</SignatureData>
```

the digest of this XML fragment is calculated as

```
3A 7B 2B 92 B3 29 11 1D B2 89 1F D2 E4 BD E7 7E
85 88 5A 38 C7 34 79 B8 1D BE FC 0F 25 1C 28 30
```

which yields the following signed info

```
<ds:SignedInfo xmlns="http://www.ebics.org/H003"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-
xml-c14n-20010315"></ds:CanonicalizationMethod>
```

```
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"></ds:SignatureMethod>
      <ds:Reference URI="#xpointer(//*[@authenticate='true'])">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"></ds:Transform>
        </ds:Transforms>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"></ds:DigestMethod>

<ds:DigestValue>OnsrkrMpER2yiR/S5L3nfoWIWjjHNHm4Hb78DyUcKDA=</ds:DigestValu
e>
      </ds:Reference>
    </ds:SignedInfo>
```

and it’s digest, which is

```
63 2D 57 CB 0F 1F 28 E7 A8 66 E9 45 F9 1E 0F 78
F1 D3 94 46 82 8B 6A 51 C8 B8 6C 46 D8 A0 BB 80
```

This digest is again completed with the ASN.1 prefix, so that finally

```
30 31 30 0D 06 09 60 86 48 01 65 03 04 02 01 05
00 04 20 63 2D 57 CB 0F 1F 28 E7 A8 66 E9 45 F9
1E 0F 78 F1 D3 94 46 82 8B 6A 51 C8 B8 6C 46 D8
A0 BB 80
```

is signed with the user’s private authentication key, resulting in the following signature:

```
97 59 9E D6 61 C7 20 01 8E B4 50 61 49 98 2F 7A
4A FF 18 75 EF 62 C9 84 9C FF 74 DE 77 9A 73 AF
8D 3C EC 3B 38 35 EC 17 AA F9 0E B3 74 EE 04 3B
2C 49 43 20 A8 14 DA 3A 69 94 36 7F F9 A5 7F AA
57 3F B4 C6 5D 1B 43 C7 13 87 6B 87 99 D8 26 51
4F C1 09 CA 54 4C F3 60 82 2D BF B1 AC 96 40 C6
DC FE EC 20 9D 0C 82 36 2E DC 99 52 B9 F3 40 30
E6 C5 1E 54 46 77 43 CE D5 A3 86 BE 2C 73 CA BE
F1 5D 5E 88 D0 3E 91 38 8C 68 E9 C0 A0 D8 DE B4
7B A1 44 90 00 E5 21 9C 81 A2 61 BF AE 8E B0 42
E7 FB E5 1C 28 6A 44 49 70 F8 A4 28 DE E2 8A E9
EF 58 39 21 A6 4F B3 D7 3C B8 FA 31 15 6A 32 92
5D 81 51 6E 74 07 7C 94 7A BB C4 FC EF F5 30 EC
4E 06 74 E0 36 0A 9D CE F5 49 55 7F B0 FE 37 A4
1F C6 F6 A1 20 DA CF 4F 97 75 1F 62 13 61 60 A8
B1 C2 B1 4D 01 16 36 66 1B EA 6F 14 A8 EF 08 76
```

Finally, the authentication signature is put into the request, which is then set to the server:

```
<ebicsRequest Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_request.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <header authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <Nonce>A5488F43223063171CA0FA59ADC635F0</Nonce>
      <Timestamp>2009-08-04T08:41:56.967Z</Timestamp>
      <PartnerID>EBICS</PartnerID>
```

```
      <UserID>EBIX</UserID>
      <Product Language="de">EBICS-Kernel V2.0.4, SIZ/PPI</Product>
      <OrderDetails>
        <OrderType>FTB</OrderType>
        <OrderID>A037</OrderID>
        <OrderAttribute>OZHNN</OrderAttribute>
        <StandardOrderParams/>
      </OrderDetails>
      <BankPubKeyDigests>
        <Authentication Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="X002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</Authentication
>
        <Encryption Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="E002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</Encryption>
      </BankPubKeyDigests>
      <SecurityMedium>0000</SecurityMedium>
      <NumSegments>1</NumSegments>
    </static>
    <mutable>
      <TransactionPhase>Initialisation</TransactionPhase>
    </mutable>
  </header>
  <AuthSignature>
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-
xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"/>
      <ds:Reference URI="#xpointer(//*[@authenticate='true'])">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"/>
        </ds:Transforms>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>OnsrkrMpER2yiR/S5L3nfoWIWjjHNHm4Hb78DyUcKDA=</ds:DigestValu
e>
      </ds:Reference>
    </ds:SignedInfo>

<ds:SignatureValue>l1me1mHHIAGOtFBhSZgvekr/GHXvYsmEnP903neac6+NPOw7ODXsF6r5
DrN07gQ7LElDIKgU2jpplDZ/+aV/qlc/tMZdG0PHE4drh5nYJlFPwQnKVEzzYIItv7GslkDG3P7
sIJ0MgjYu3JlSufNAMObFHlRGd0PO1aOGvixzyr7xXV6I0D6ROIxo6cCg2N60e6FEkADlIZyBom
G/ro6wQuf75RwoakRJcPikKN7iiunvWDkhpk+z1zy4+jEVajKSXYFRbnQHfJR6u8T87/Uw7E4Gd
OA2Cp3O9UlVf7D+N6QfxvahINrPT5d1H2ITYWCoscKxTQEWNmYb6m8UqO8Idg==</ds:Signatu
reValue>
  </AuthSignature>
  <body>
    <DataTransfer>
      <DataEncryptionInfo authenticate="true">
        <EncryptionPubKeyDigest
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Version="E002">dFAYe281vj9NB7w+VoWIdfHnjY9hNbZLbHsDOu76QAE=</EncryptionPubK
eyDigest>

<TransactionKey>ShMnR//KeC/3nKGzN3RkYT8UhUSFp4YXkjDZJ0/GOE8F1165JdN/HJAHyuc
5tiSalI3ayT9X1XYtBQW5W0X/CBeXJP5BJNzXWuPJWINGfQ9NKnWnT8pNtvI0PpvU8Bbh+xv4+E
tVCoklQdurqJS23v/1VgdnBF9FGQVsLSYQvkM=</TransactionKey>
      </DataEncryptionInfo>
      <SignatureData
authenticate="true">yuSeXqH2iy44fj9tokf9JzPE1y7VnPrBIv3o0enAiQ+oP2nEFjyX5Ch
XZDeuVcoEQB9dlKMxS7YUQ64NOuzkUTKccy7ZF79Ctd9on+fjg9e/utO26gZh2+JPtycOUYlpfN
```

```
oUuaA2No5JYMehUR8x43iFpw1vCT3bVqVuCxjbxJVHES6kf9kQMMbAEk96vY1Fzart3yLDVTFfX
gkqHowP42FHPdFKjSKKhkHm2poGD9N9jtCuh+SU+52A02zBTUR/1wADGIzPScDdNE1sm85e+XEt
FhuBKI9E5TRb5HQ9Ne1RG0P9HAWFBwGgls1dbAoD3bMxkZkvdeResJipbIp99NitonCA8bJQ3Is
YwN8RXQSul1FhiFcbqGUW575/QHE9YYoHX9Zjp8lt8xHDyq10cSulWW6T+1x06x63ySARrXQWcM
VyF5ukffJqvb3PECP9CED5W4SZ9mA/GqPTMLgQQmwbn4TZkY5yU0TdUw8edjM6nxyXeZpJ42Jiv
Lm1Bwi/lRYzzyR+P/qrUsYgQQM+tXrdzAXi0X9fZfkmIhTeg2ScmKRhgcHncThw7SvdxD9t9KoL
IrlouR6cltJp6xdGCxCnSs8JjjX6jY0+uzriahYH+t4nakGapwswSyDoq4VkGreLQCeSIBxYzI7
+dZ7UUUZS653otxb/Uho2ReDiwfRjqRB7XU+1vqGAL9N6gEOB</SignatureData>
    </DataTransfer>
  </body>
</ebicsRequest>
```

From the server, we get the following request, indicating that everything worked well.

```
<?xml version="1.0" encoding="UTF-8"?>
<ebicsResponse Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_response.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <header authenticate="true">
    <static>
      <TransactionID>ABF70F5E704E03CFE5B50692BEE55ABA</TransactionID>
    </static>
    <mutable>
      <TransactionPhase>Initialisation</TransactionPhase>
      <ReturnCode>000000</ReturnCode>
      <ReportText>[EBICS_OK] OK</ReportText>
    </mutable>
  </header>
  <AuthSignature>
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-
xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"/>
      <ds:Reference URI="#xpointer(//*[@authenticate='true'])">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"/>
        </ds:Transforms>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>0/3J0Cne3ncZmTUNZuD8S8HOKI95bo9fP1AvVyVZDfc=</ds:DigestValu
e>
      </ds:Reference>
    </ds:SignedInfo>

<ds:SignatureValue>J1vtnW2TX3sY9IWnWDcYA2SiZcPwSKq8ap7ESJEfHjDsqDhJV5A3dOo/
R4v3LwoMe7CdbWc0AX+KMrcL4FSeQ+CnM776ncQV3OELS83ZkHZzbp15B0cJ9pf3lY9JiEDxKLM
eoRjNJlc97Io2iJOW6dO2ZoSPXjF2G4ukQ81P7P8=</ds:SignatureValue>
  </AuthSignature>
  <body>
    <ReturnCode authenticate="true">000000</ReturnCode>
    <TimestampBankParameter authenticate="true">2009-06-
16T17:56:05.698+02:00</TimestampBankParameter>
  </body>
</ebicsResponse>
```

The signature verification is done just the same way as with HPB. We retrieve the transaction id from the response for further use:

```
AB F7 0F 5E 70 4E 03 CF E5 B5 06 92 BE E5 5A BA
```

Transfer phase

We will now send the first data segment, which is in fact also the last. The order data is compressed, encrypted with the transaction key and base64 encoded before it's incorporated into the XML request. Note that the following XML does not contain all the order data, as this would be about 50000 characters long. You will find the complete request file in the XML examples. The transaction id is taken from the initialisation response.

```
<ebicsRequest Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_request.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <header authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <TransactionID>ABF70F5E704E03CFE5B50692BEE55ABA</TransactionID>
    </static>
    <mutable>
      <TransactionPhase>Transfer</TransactionPhase>
      <SegmentNumber lastSegment="true">1</SegmentNumber>
    </mutable>
  </header>
  <AuthSignature/>
  <body>
    <DataTransfer>
      <OrderData>R47GR7NB... axOg==</OrderData>
    </DataTransfer>
  </body>
</ebicsRequest>
```

Note that neither the body tag not the DataTransfer tag or the OrderData tag has the authenticate attribute set to true. The actual order data is therefore not X00 signed, the integrity of the order data is guaranteed by the A00 signature.
Processing this request in the way we have already seen twice, we get the following signed request (parts of the order data are again omitted):

```
<ebicsRequest Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_request.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <header authenticate="true">
    <static>
      <HostID>SIZBN001</HostID>
      <TransactionID>ABF70F5E704E03CFE5B50692BEE55ABA</TransactionID>
    </static>
    <mutable>
      <TransactionPhase>Transfer</TransactionPhase>
      <SegmentNumber lastSegment="true">1</SegmentNumber>
    </mutable>
  </header>
  <AuthSignature>
    <ds:SignedInfo>
```

```
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-
xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"/>
      <ds:Reference URI="#xpointer(//*[@authenticate='true'])">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"/>
        </ds:Transforms>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>Z9ULhiRFz40rJJIKV6G16TH3z7pC/Qy/BCKstjtA/UQ=</ds:DigestValu
e>
      </ds:Reference>
    </ds:SignedInfo>

<ds:SignatureValue>c9P6ESQZnnFhe+CQ0iMyiAJO0GePdm2f3y0Byxp6ACLLW31UYoAMnrgq
BFDXxgLpA1qQfxmgNL0VwExuw9W7UKYBzcj0TCMOoUjL4FKEcCGOXh+dU2Kef6GTKaF3/05vjqZ
/CHCjHlK9Bne+czmOQb3nSqREjtgSqcMFkLSNjwCm5Sf43l35dZsQ3lLZlU+GHS4Hlo46f/38Ta
OaYO0TbkAd+f4Qw/u6K92NRWxQcj4WuEnCj5rCm5oNknnGCj1wMOs+shsM6uATeniS46jJVQLaq
uBqXurLchGORdMMPajc+oxOcQvdCYe87r1z5ugrFMINOyKnUcobzkvyQsr1cw==</ds:Signatu
reValue>
  </AuthSignature>
  <body>
    <DataTransfer>
      <OrderData>R47GR7NB... axOg==</OrderData>
    </DataTransfer>
  </body>
</ebicsRequest>
```

The servers responds with the following message:

```
<?xml version="1.0" encoding="UTF-8"?>
<ebicsResponse Revision="1" Version="H003"
xsi:schemaLocation="http://www.ebics.org/H003
http://www.ebics.org/H003/ebics_response.xsd"
xmlns="http://www.ebics.org/H003"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <header authenticate="true">
    <static>
      <TransactionID>ABF70F5E704E03CFE5B50692BEE55ABA</TransactionID>
    </static>
    <mutable>
      <TransactionPhase>Transfer</TransactionPhase>
      <SegmentNumber lastSegment="true">1</SegmentNumber>
      <ReturnCode>000000</ReturnCode>
      <ReportText>[EBICS_OK] OK</ReportText>
    </mutable>
  </header>
  <AuthSignature>
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-
xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"/>
      <ds:Reference URI="#xpointer(//*[@authenticate='true'])">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"/>
        </ds:Transforms>
```

```
        <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>80FslBHne4Bd3MbM8QzZLBLQUgMUTSSk4ODLkZaiq1U=</ds:DigestValu
e>
      </ds:Reference>
    </ds:SignedInfo>

<ds:SignatureValue>SBW+TvhL+FPRAxsMxGk4I1DP5OrrH+7scQQrYFTnns74XCv94eGWES6W
8Yws783Dqkjbmmr2+t9SW2Pr/+GM+FG0RYb0/7AAF/+JRbzXurWIqXCEGHOz/tHoC/GYll+AgYu
+5zCfplaYSW2012tJrghHaJoKJJnMgCQzEZhfcs0=</ds:SignatureValue>
  </AuthSignature>
  <body>
    <ReturnCode authenticate="true">000000</ReturnCode>
    <TimestampBankParameter authenticate="true">2009-06-
16T17:56:05.698+02:00</TimestampBankParameter>
  </body>
</ebicsResponse>
```

After validating the X00 signature and considering the return codes, we know that our file was successfully uploaded to the EBICS server.

Sending a FUL order with Certificates

INI

HIA

HPB

Upload