DIN NI-17.4, TeleTrusT AG2
Editor: Bruno Struif

German comments to PKCS#15, Version 1.0 (incomplete draft)

1. Clause 3, AID, replace application provider number by application provider identifier

   Reason: ISO conformance

2. Clause 3, APDU, replace host computer by interface device e.g. host computer.

   Reason: The current wording is too restrictive.

3. Clause 3, Cardholder, replace presenting a smart card for use by for whom the card was issued.

   Reason: The current definition is invalid.

4. Clause 3, Card issuer, replace owns and provide a smart card product by issues the related cards.

   Reason: The current definition is invalid, e.g. the cardholder may be the owner of the card.

5. Clause 3, CHV, replace the second sentence by: A knowledge based item (PIN or password) or a biometrical template. The wording ´Typically ..´ should be moved to PIN.

   Reason: The current wording is too restrictive.

6. Clause 3, Cryptogram, add at the end: to provide confidentiality.

   Reason: The current wording is incomplete.

7. Clause 3, EF, replace identifier by file identifier

   Reason: ISO conformance

8. Clause 3, Memory card, add at the end: and possibly hardwired security functions.

   Reason: The current wording is incomplete.

9. Clause 3, Message, replace internal device by interface device

   Reason: Correction.

10. Clause 3, Password, insert behind data ´or functions´.

    Reason: The current wording is incomplete.

11. Clause 3, PIN pad, replace alphanumeric and command key s by (alpha-)numeric and function keys

    Reason: Correction.

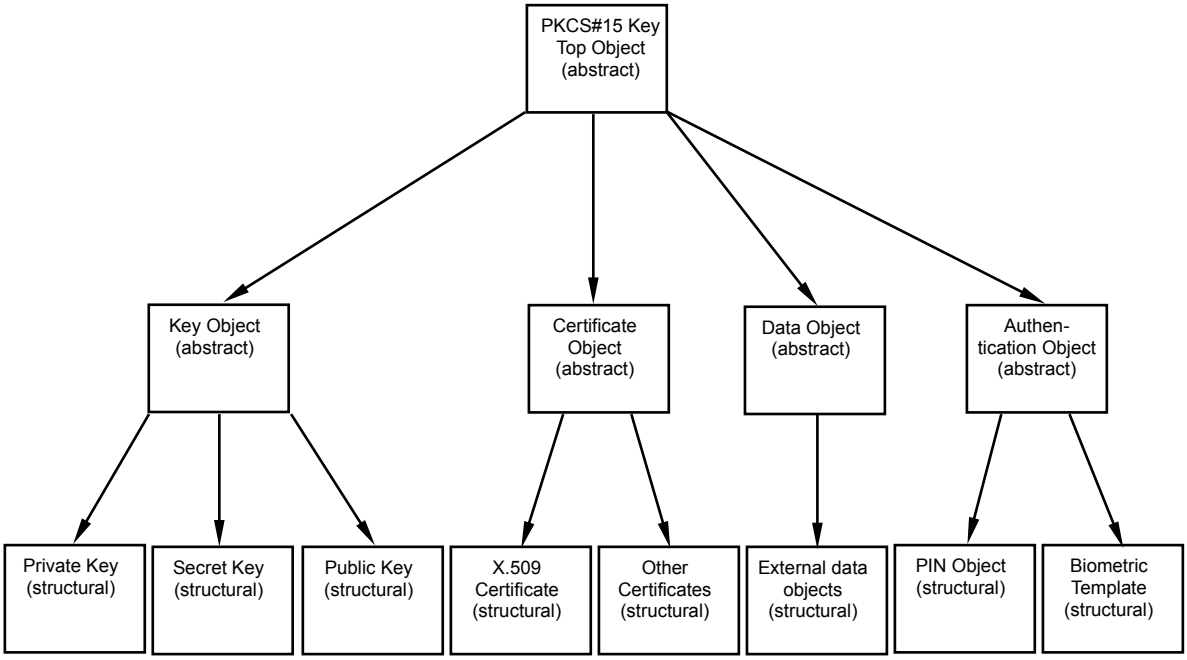
12. Clause 3, Provider, erase this definition.

    Reason: This definition is not really helpful. Application provider and card issuer are already defined.

13. Clause 4, Symbols and Abbreviations, add all the missing abbreviations like AODF, CDF, PrKDF,PuKDF, DODF, PGP, WTLS etc.

    Reason: Completion.

14. Clause 5.1.1, Object classes, replace fig. 1 by the following figure.



Reason: Authentication objects of the class ´biometric template´ should be added, since it is now feasible to run feature matching algorithms e.g. for voiceprint and fingerprint in the card.

15. Clause 5.1.3, Access methods, add at the end of the second sentence: or key objects.

    Reason: In a card there may be files protected e.g. by a cryptographic authentication procedure. Example: a physician has to prove its access rights within a symmetric or asymmetric authentication procedure.

16. Clause 5.1.3, Access methods, replace in the 3rd sentence PINs by: knowledge-based or biometrical user authentication.

    Reason: Completion

17. Clause 6.1, Overview, 2nd paragraph, 1st sentence, replace bracket by: (ISO/IEC 7816-4 and possibly ISO/IEC 7816-8 and -9)

    Reason: Completion

18. Clause 6.2 IC card requirements, add at the end: Extended features may require to support ISO/IEC 7816-8 and -9 or part of them.

    Reason: Completion

19. Clause 6.3 Card File Structure, add the follwoing figures:

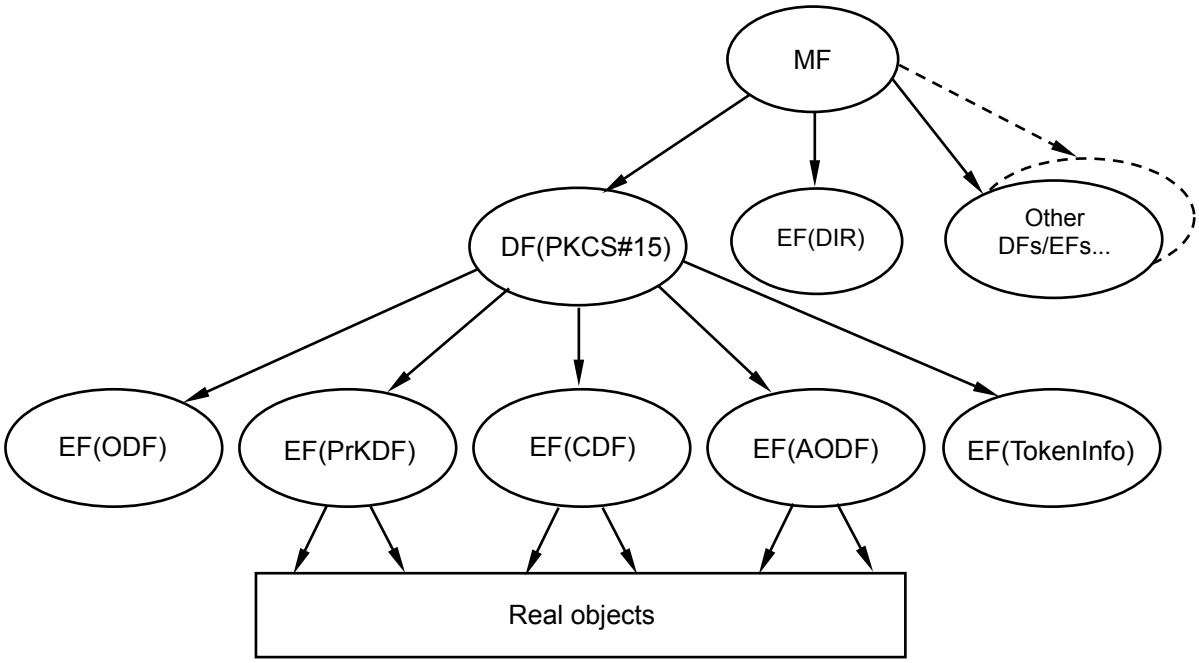


Figure 4: Contents of DF(PKCS#15) and other DFs (Example)

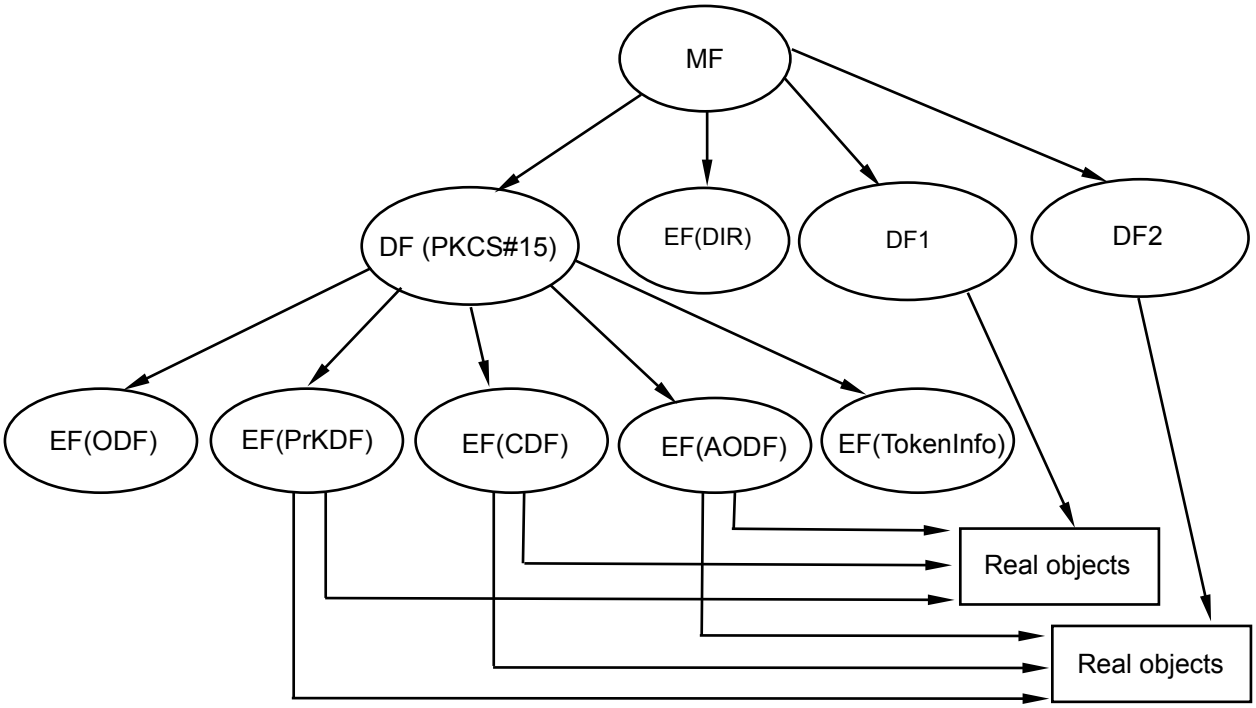


Figure 5: Contents of DF(PKCS#15) and other DFs (Example)

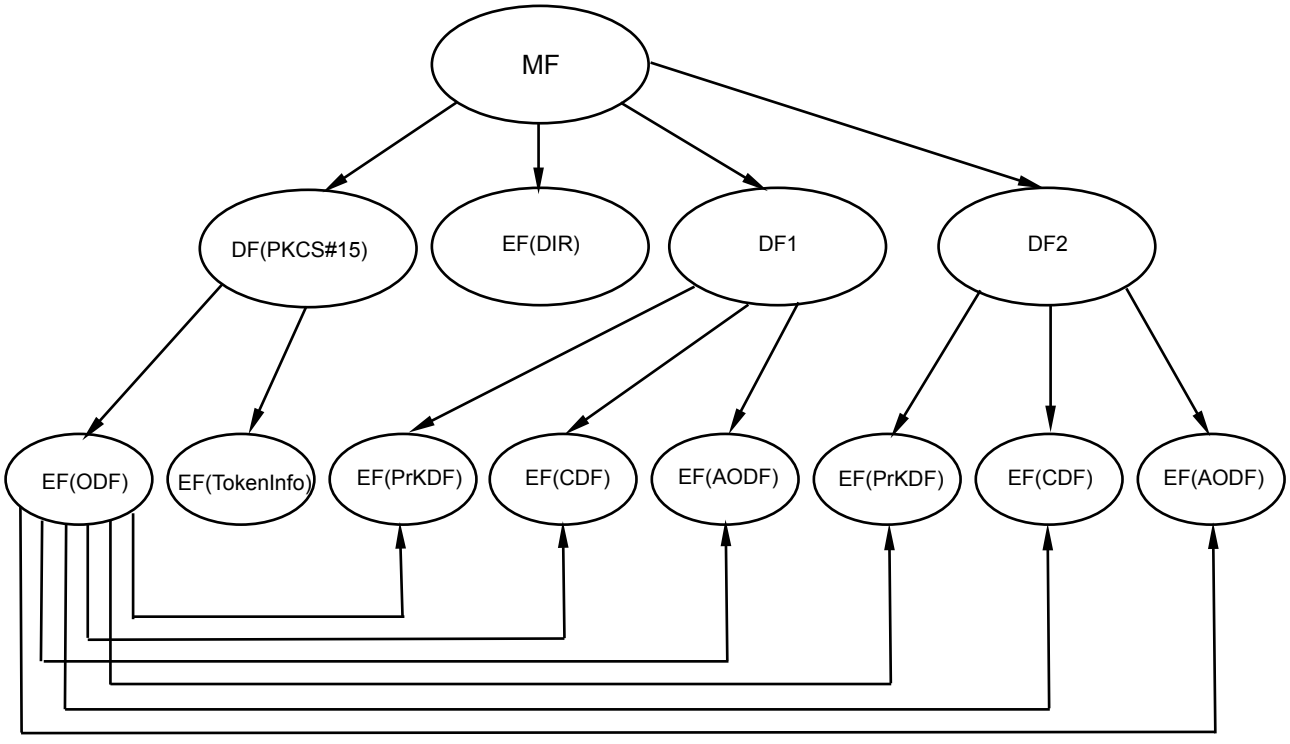


Figure 5: Contents of DF(PKCS#15) and other DFs (Example)

Reason: The current clause 6.3 is not clear enough.

20. Clause 6.5.7 AODFs, replace in the first line e.g. PINs by PINs, passwords, biometric data

    Reason: It is worth to mention the relevant authentication objects

21. Clause 6.7.1, replace registered identifier by registered application provider identifier

    Reason: ISO conformance

22. Clause 7.1.3 PKCS15CommonCertificateAttributes, insert in the syntax definition below authority the following line:
    keyUsage KeyUsage OPTIONAL,

    Reason: In a smartcard there may be certificates for non repudiation, digital signature, key encipherment etc. To be able to select the certificate needed, the key usage field as defined for X.509 certificates is necessary.

23. Clause 7.1.3 PKCS15CommonCertificateAttributes, replace thumbprint by digest or another appropriate word.

    Reason: The meaning of a thumbprint in this context is that one of a hash value. Since thumbprint is also a biometric authentication object, it cannot be used here for something else.

24. Clause 7.2 PKCS15Objects type, add the following object type in the list:

    cvCertificates [9] PKCS15Certificates,

    and the following paragraph at the end:

    The **cvCertificates** field shall point to card verifiable certificates issued to the card and/or the cardholder. CV certificates (see ISO/IEC 7816-8) can be processed within a smartcard or a smartcard-like security module and play therefore an important role in asymmetric authentication procedures.

    Reason: This functionality is urgently needed.

25. Clause 7.6 PKC15Certificates Type, insert a further choice:

    cvCertificate [5] PKCS15CertificateObject {PKCSCVCertificateAttributes},

    Reason: Consequence of comment 24.

26. Clause 7.66, add behind a further clause:

    7.6.7 CV certificate objects

    PKCSCVCertificateAttributes ::= SEQUENCE {
    value PKCSObjectValue { PKCS-OPAQUE.&Type},
    ... -- For future extensions
    }

The semantics of the fields is as follows:

PKCSCVCertificateAttributes.value: The value shall, in the IC card case, be a **`pkcs15ReferencedValue`** identifying a file containing a cv certificate.

27. Clause 7.8, add a further choice:

    bio PKCS15AuthenticationObject { PKCSBioAttributes },

    and erase the words behind future extensions.

    Reason: There are already several implementations of biometric feature matching algorithms in cards, e.g. for voiceprint and fingerprint.

28. Clause 7.81 Pin Objects replace in the line pinReference the word OPTIONAL by DEFAULT 0,

    Reason: PINs have always a reference in the card. Reference 0x00 means "implicitly selected" and is used, if no specific reference is required.

29. Clause 7.8.1 Pin Objects, bullet unblockingPin, replace the current wording by

    unblockingPin (ISO/IEC 7816-8: resetting code), meaning that this authentication object is used for unblocking purposes, i.e. to reset the retry counter of the related authentication object to its initial value

    Reason: ISO conformance

30. Clause 7.8.1 Pin objects, add two further PinFlags:

    authentic (9),
    enciphered (10),

    add under semantics:

    - **authentic** means, that the PIN is presented as an SM data object (plain value DO or cryptogram DO depending on the enciphered flag) with a cryptographic checksum DO
    - **enciphered** means, that the PIN is presenetd as a cryptogram DO

    add under abbreviations:
    SM = Secure Messaging

    Reason: The PIN is not always presented as plain value.

31. Clause 7.8.1.1 Transforming a supplied PIN

    - Better title: PIN formats
    - replace in 1. the word Convert by Present
    - replace in b) and c) the word verify by encode the PIN in such a way
    - insert d) If other formats are used at the interface to a card (e.g. the banking format 2

PIN block), then this special encoding has to be added to the PKCS15PinType and the PIN has to be encoded according to this type

Reason: Completion and better understanding

32. Clause 7.8.1.1, add behind the new clause 7.8.2:

7.8.2 Bio Objects

```
PKCS15BioAttributes  ::= SEQUENCE {
        bioFlags          PKCS15BioFlags,
        bioSubject        PKCS15BioSubject,
        bioType           PKCS15BioType,
        bioReference      [0] PKCSReference DEFAULT 0,
        lastBioChange     GeneralizedTime  OPTIONAL,
        path              PKCS15Path OPTIONAL,
        ...  -- For future extensions
        }

PKCS15BioFlags  ::= BIT STRING {
        reserved          (0),
        local             (1),
        change-disabled   (2),
        unblock-disabled (3),
        initialized       (4),
        reserved          (5),
        reserved          (6),
        reserved          (7),
        diable-allowed    (8),
        authentic         (9),
        enciphered        (10),
         }

PKCSBioSubject  ::= CHOICE {
        fingerPrint           [0] FingerPrint,
        voicePrint            [1] VoicePrint,
        irisPrint             [2] IrisPrint,
        facePrint             [3] FacePrint,
        retinaPrint           [4] RetinaPrint,
        handGeometry          [5] HandGeometry,
        writeDynamics         [6] WriteDynamics,
        keystrokeDynamics     [7] KeystrokeDynamics,
        lipDynamics           [8] LipDynamics,
        ... -- For future extensions
        }

FingerPrint ::=  SEQUENCE {
         handID      HandID,
         fingerID    FingerID
         }
```

HandID ::= ENUMERATED {righthand (0), lefthand (1) }

FingerID ::= ENUMRATED { thumb(0), pointer finger (1), middle finger (2), ring finger (3), little finger (4) }

-- the others to be worked out

PKCSBioType ::= OBJECT IDENTIFIER

- PKCS15BioAttributes.bioFlags: see pinFlags, but replace PIN by BRD (Biometrical Reference Data)

- PKCS15BioAttributes.bioSubject: This field determines the biometrical subject, e.g. the pointer finger of the right hand.

- PKCS15BioAttributes.bioType: This field determines which biometrical data structure has to be presented to the card.

33. PKCS #15 describes objects residing in the cryptographic token. However, the general problem is, that the interface device must know how to use them at the interface to the card, i.e. to the static object description a service desciption has to be added which makes clear, which command sequence has to be send to the card to achieve the respective service. The DIN Security Service Descriptor concept, which is an extended version of the SSD concept worked out originally in the European project TrustHealth by Swedish and German experts, is a method to cover the identified gap. The SSD concept is presented in the annex.

34. Topics not commented until now:

    - secure messaging
    - authentication procedures
    - different authentication requirements (once, each time for using a special service e.g. digital signature
    - ...

Annex

Security Service Descriptor Templates

1 Security Service Descriptor Concept

For supporting interoperability and coexistence of chipcards with differences, e.g. in command sequences, as well as to facilitate migration in an easier way, an SSD file should exist in the SigG application. The information about available security services are provided in SSD templates to be interpreted by the terminal. The SSD file contains either

- One or more SSD-Templates for each security service (see Table F.1) or
- a DO ´SSD Profile identifier´ (see DO with tag ´8D´).

2 SSD Data Objects

The SSD templates have context-specific tags, whereby the context is given by the SSD file. The DO ´Instruction set mapping´ is mandatory in the value part of each SSD template. All other DOs are optional.

Tag	**Meaning**	**Related ISO/IEC commands** (commands in ( ) optional)
´A0´	User authentication service	VERIFY or CHANGE RD or ENABLE VR or DISABLE VR or RESET RC
´A1´	Internal authentication service	INT. AUTHENTICATE
´A2´	External authentication service (sym.)	GET CHALLENGE EXT. AUTHENTICATE
´A3´	External authentication service (asym.)	(MANAGE SE) PSO: VERIFY CERTIF. GET CHALLENGE EXT. AUTHENTICATE
´A4´	Digital signature computation service	(MANAGE SE) (PSO: HASH) PSO: COMPUTE DS
´A5´	Digital signature verification service	(MANAGE SE) (PSO: HASH) PSO: VERIFY DS
´A6´	Certificate verification service	(MANAGE SE) PSO: VERIFY CERTIFICATE
´A7´	Checksum computation service	(MANAGE SE) PSO: COMPUTE CC
´A8´	Checksum verification service	(MANAGE SE) PSO: VERIFY CC
´A9´	Encipherment service	(MANAGE SE) PSO: ENCIPHER
´AA´	Decipherment service	(MANAGE SE) PSO: DECIPHER
´AB´	File management service	SELECT FILE

Table 1: SSD Templates

- DO Instruction set mapping (ISM), tag ´80´
  This DO contains the regular command to provide the respective security service. If the security service requires a sequence of commands, then this DO is repeated. The DO contains at least the CLA-byte and the INS-byte of the command as defined in main part of this specification. If appropriate, P1 and P2 or

further parts of the command APDU may be present. It is assumed that the outside world knows the command in its complete form and which data - if any - have to be inserted in the body part of the command.

- DO Command to perform (CTP), tag ´52´ (see ISO/IEC 7816-6)
  This DO - if present - informs the outside world which command is supported by the card to provide the same security service as achievable with the command indicated in the DO ISM. If several commands are necessary then the DO CTP is repeated. It shall follow - if used - immediately behind the last DO ISM.

- DO Algorithm object identifier (OID), tag ´06´ (see ISO/IEC 7816-6)
  The value field of this DO contains the object identifier of the related crypto algorithm available in the card. The encoding rule of the OID is according to ISO 8825 as follows:

  - first subidentifier is multiplied by 40 and coded as binary number in one byte; the value of the second subidentifier is added to this byte
  - the following subidentifiers are each represented as a binary number in a sequence of bytes from which bit b8 is the chaining bit
  (1 = not last byte, 0 = last or only byte)

- DO Algorithm reference, tag ´81´
  The value field of this DO contains the algorithm reference as used in the card for the algorithm denoted by the DO OID or specified in an application context.

- DO Key reference, tag ´82´
  The value field contains the key reference of the key to be applied by the card when performing the related security operation. If this DO is present, then the value has to be used in the command related to this security service.

- DO FID key file, tag ´83´
  The value field contains the file id of a file containing the key to be applied by the card when performing the related security operation. If this DO is present, a SELECT FILE command with the respective FID has to be sent as first command before using a security service.

- DO Key group, tag ´84´
  This DO is only relevant for symmetric encipherment algorithms which work with individual keys and master keys. When this DO is used, then the value denotes the entity using the master key (e.g. ´01´ = physician, ´02´ = pharmacist, i.e. the values can be considered as group ids)

- DO FID base certificate file, tag ´85´
  The value field of the DO contains the file id of a file containing the base certificate related to the respective security service.

- DO FID adjoint certificate file, tag ´86´
  The value field contains the file id of a file containing an adjoint certificate related to the base certificate of the same security service template.

- DO Certificate reference, tag ´87´
  If no FID for a certificate file is given, then this DO - if present - contains a reference to the related certificate which is not stored in the card. The reference may be a key identifier (tag ´88´) and/or the distinguished name (or the ID) of the CA issuing the certificate (tag ´89´) followed by the certificate serial number (tag ´8A´).

- DO Certificate qualifier, tag ´88´
  The value field contains the information whether the certificate is a non-ICC certificate (e.g. a X.509 certificate) to be verified outside a card (value ´00´) or an ICC certificate, which may be interpreted by a card (value ´01´). The default value is ´00´.

- DO FID for file with public key of the certification authority PK(CA), tag ´89´
  This DO - if present - contains the FID of the file in which the DO public key of the certification authority is stored (tag ´5F4A´).

- DO PIN usage policy, tag ´5F2F´ (see ISO/IEC 7816-6)
  This DO - if present - indicates the PIN usage policy. The content of this DO is application specific.
  For the SigG application the following values are defined:
  ´00´: after PIN presentation no limitation
  ´01´-´0F´: the number indicates the amount of signatures possible with one PIN presentation
  ´10´-´FF´: RFU

- DO PIN reference, tag ´8A´
  The value field of this DO contains one byte coding the qualifier of the reference for the cardholder verification data, if the default value ´00´ is not used.

- DO Application identifier (AID), tag ´4F´ (see ISO/IEC 7816-6)
  This DO indicates, to which application the related template belongs and shall only be used, if an SSD file on the MF level is needed (e.g. if a PIN presentation before selection of the application with the specified AID is required).

- DO CLA coding, tag ´8B´
  The value field of this DO contains the CLA coding which has to be used instead of the CLA coding given in the DO ISM.

- DO Status information (SW1-SW2), tag ´42´ (see ISO/IEC 7816-6)
  If relevant status bytes (e.g. from the VERIFY command) differ from those of the command described in the DO ISM, then the mapping shall be given, i.e. a SW1-SW2 of the ISM command is followed by SW1-SW2 coding send by the card.

- DO Discretionary data, tag ´53´ (see ISO/IEC 7816-6)
  The contents of this DO - if present - is defined by the application provider.

- DO SE number, tag ´8C´
  The value field of this DO contains the security environment number.

- DO SSD profile identifier, tag ´8D´
  The value of this DO identifies for the respective application an SSD set available in the IFD. The externally stored SSD set describes the security service supported by the card.

- DO FID mapping, tag ´8E´
  The value of this DO contains one or more pairs of FIDs: the first FID gives the value assigned in this specification, the second FID indicates the FID to be used.

Template	**Meaning**
A0-06- [80-04-0020008106]	VERIFY for knowledge based user authentication
A0-06- [80-04-00240081]	CHANGE RD
A0-06- [80-04-002C0081]	RESET RC *)
A4-11- [80-04-002A9E9A || 81-01-01 || 85-02-C000 || 86-02-C008]	PSO: COMPUTE DS; AlgID = ´01´ (= RSA with DSI acc. to ISO/IEC 9796-2); FIDs of the certificate files EF.C.CH.DS and EF.C.CA.DS **)
*) Template is missing, if the service is not supported **) DO FID is missing, if the certificate is not on the chipcard	

Table 2: Example of an SSD file