Contactless authentication protocols for Machine Readable Travel Documents (MRTDs)

Dr. Kim Nguyen

Bundesdruckerei GmbH, Berlin, Germany

CHES 2007, Vienna, 2007-09-10

Agenda

- **Introduction to MRTDs and international standardization**
- **Main threats**
- **Passive Authentication**
- **Contactless Authentication protocols**
  - Without infrastructure: BAC
  - With Public key infrastructure: EAC
- **Discussion**

MRTDs

- Machine Readable Travel Document (MRTD):

  Official Document issued by a State or organization which is used by the holder for **international travel** (e.g. passport, visa, official document of identity) and which contains **mandatory visual** (**eye readable**) data and a seperate mandatory data summary in a format which is **capable of being read by machine**.

- „Conventional“ passports are already machine readable:
  they have a MRZ (=machine readable zone)

Integration of biometric features in MRTDs (EU)

- **ICAO documents regarding MRTDs published (including BAC): October 1st, 2004**
- **EU Council decision of December 12th, 2004**
- **Introduction of ePassport in Germany November 2005**
  - Phase 1: facial image as primary identification feature
- **All EU member countries should issue biometric passports by August 2006**
- **EU Council decision of June 28th, 2006:**
  - Phase 2: Fingerprints must be protected using Extended Access Control.
- **Introduction of EAC ePassport in Germany November 2007**
  - Phase 2: inclusion of fingerprints (EAC protected) as secondary identification feature

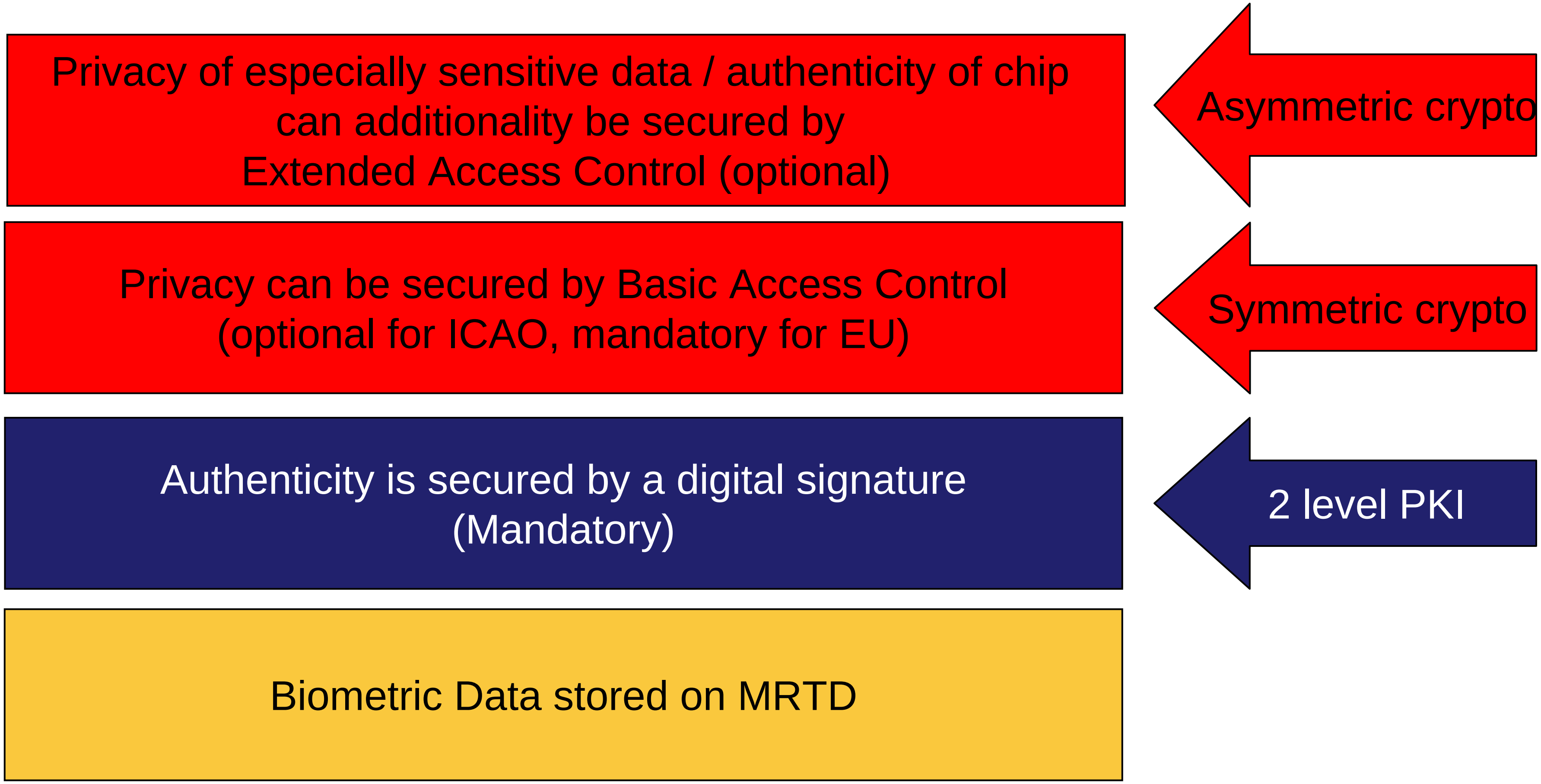
Biometric features in MRTDs

- **Content if information to be stored is determined by ICAO and EU specifications**
  - LDS (=logical data structure) contains
    - Digitized machine readable zone (MRZ)
    - Facial image
    - Fingerprints (in phase 2)
    - Digital signature
- **Storage medium is determined by ICAO and EU specifications**
  - Contactless chip with non-volatile memory of at least 32 kB
  - Contactless interface according to ISO 14443 (type A or B)
- **Interoperability will be guaranteed by conformity testing of contactless interface, OS behaviour and data structure:**
  - Specifications are currently finalized by ISO / ICAO

Main threats to biometric enabled MRTDs

- **Data exchange (write new self-generated data onto chip):**
  - Prevented by HW and SW countermeasures (read-only configuration)
- **Authenticity of data**
  - Secured by means of a digital signature (two-level PKI)
- **Confidentiality of data:**
  - Secured by means of authentication protocols and encrypted data transfer
- **Tracking:**
  - Prevented by means of authentication protocols
  - Also random contactless UID is required

Security Mechanisms



Privacy of especially sensitive data / authenticity of chip can additionality be secured by Extended Access Control (optional)

Asymmetric crypto

Privacy can be secured by Basic Access Control (optional for ICAO, mandatory for EU)

Symmetric crypto

Authenticity is secured by a digital signature (Mandatory)

2 level PKI

Biometric Data stored on MRTD

ICAO compliant signature algorithms

- **ICAO PKI report, version 1.1 offers three choices for document signing:**
  - RSA (using at least 2048 bit modulus)
    - RSA PSS-SSA is recommended
  - DSA (using ground field of at least 2048 bits)
  - ECDSA (order of base point at least 224 bits)
    - No restrictions on base field given
    - German ePassport uses ECDSA over GF(p)
- **Hash functions currently in use:**
  - SHA-1, SHA-256 for RSA based algorithms
  - SHA-1 for ECDSA (lack of standardized algorithms using other hash functions -> ISO 15496 and BSI TR 3111)

Contactless authentication protocols

- **Two types of authentication protocols are implemented for MRTDs:**
  - Authentication protocol without need for infrastructure
    - Symmetric (3DES based) protocol
    - Protocol initialisation only requires information that is optically available on the passport
    - => Basic Access Control (BAC)
  - Authentication protocols with need for infrastructure
    - Asymmetric protocol
    - Protocol requires access to certificates for terminals (PKI)

BAC: Introduction

- **Authentication mechanism to protect basic identification feature (facial image) and personal data in the MRZ**

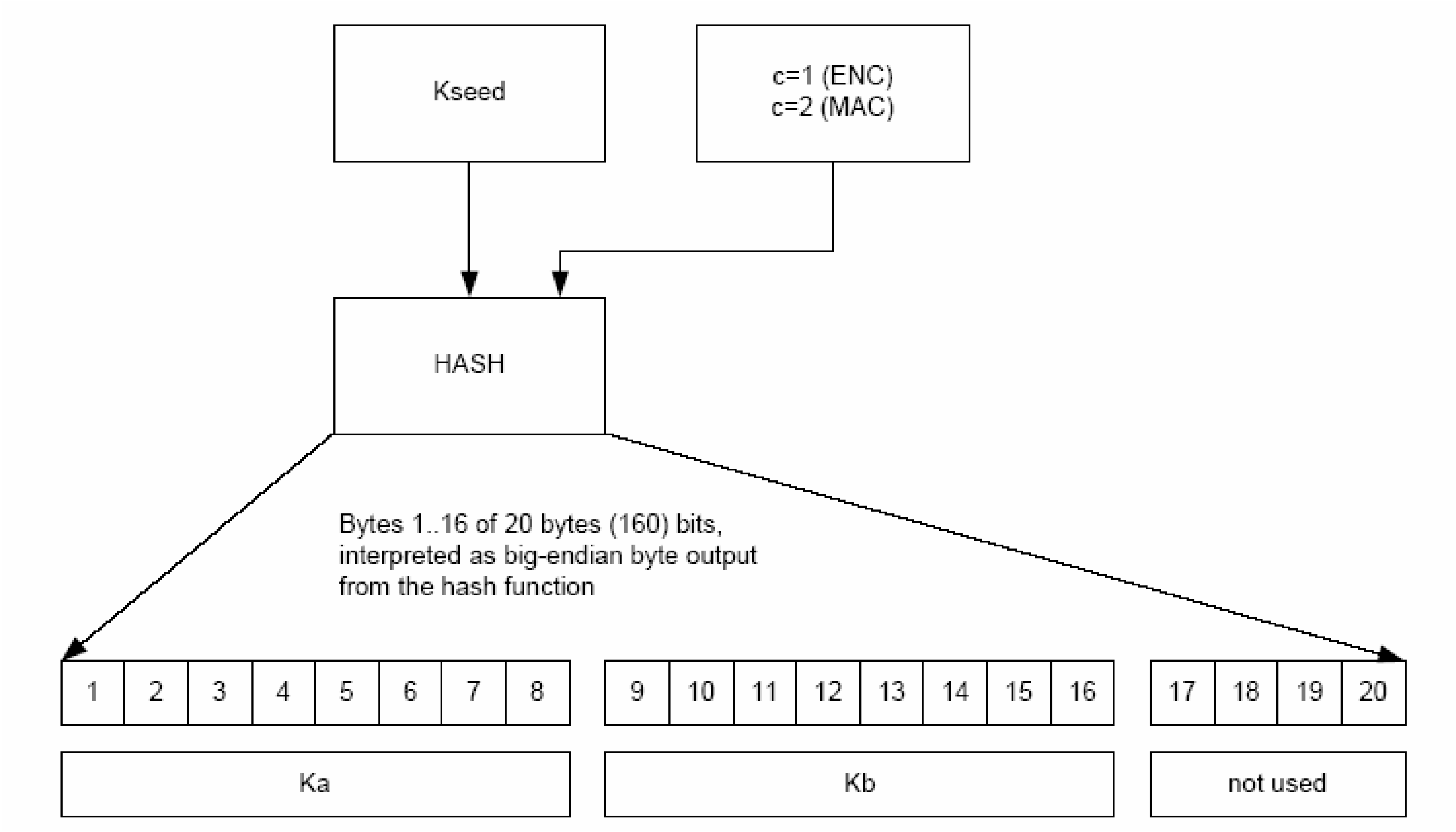
- **Based on optical information printed on passport, hence no infrastructure needed to perform the protocol.**

- **Intended to prevent skimming (reading out of information via contactless information without consent of passport holder) and eavesdropping (no plain information is sent over the contactless interface)**

BAC 1/3

- key seed K_{seed}:
  - derived either from printed MRZ (passport serial number, date of birth, expiry date, only readable when pass is open) or
  - derived during mutual authentication (session keys)
- 3DES keys for each session derived from K_{seed} (each with 2 keys):
  - K_{ENC} for encryption
  - K_{MAC} for building Message Authentication Codes (MACs)
- 32 bit counter c allows to chose the keys to be derived from K_{seed}:
  - c = ‘0x 00 00 00 01’ for K_{ENC}
  - c = ’0x 00 00 00 02’ for K_{MAC}

BAC 2/3



Kseed

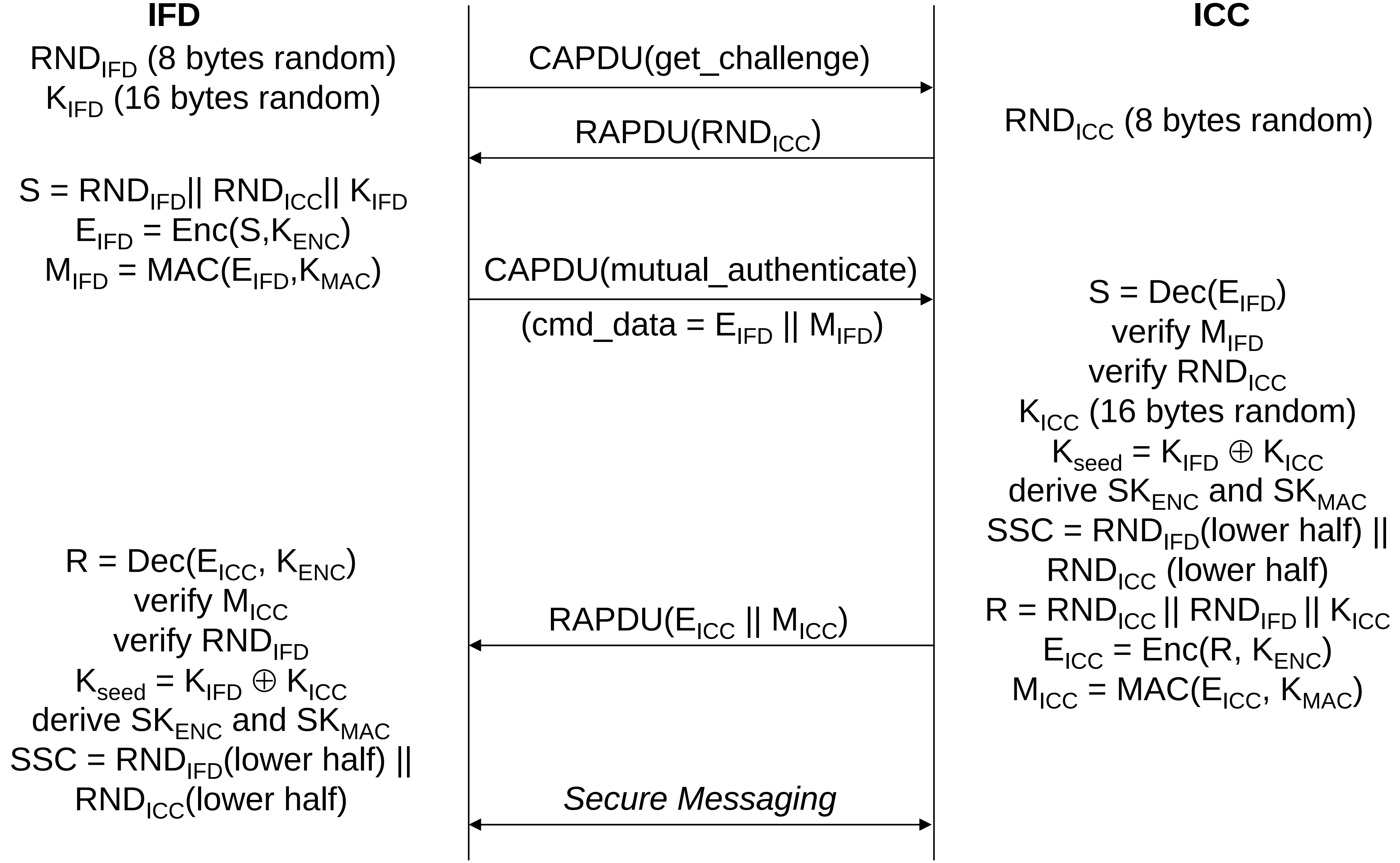
c=1 (ENC)
c=2 (MAC)

HASH

Bytes 1..16 of 20 bytes (160) bits,
interpreted as big-endian byte output
from the hash function

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Ka								Kb								not used			

BAC 3/3



IFD

RND_{IFD} (8 bytes random)
K_{IFD} (16 bytes random)

$S = RND_{IFD} || RND_{ICC} || K_{IFD}$
$E_{IFD} = Enc(S, K_{ENC})$
$M_{IFD} = MAC(E_{IFD}, K_{MAC})$

$R = Dec(E_{ICC}, K_{ENC})$
verify M_{ICC}
verify RND_{IFD}
$K_{seed} = K_{IFD} \oplus K_{ICC}$
derive SK_{ENC} and SK_{MAC}
$SSC = RND_{IFD}$(lower half) ||
RND_{ICC}(lower half)

CAPDU(get_challenge) →

← RAPDU(RND_{ICC})

CAPDU(mutual_authenticate) →
(cmd_data = E_{IFD} || M_{IFD})

← RAPDU(E_{ICC} || M_{ICC})

↔ *Secure Messaging*

ICC

RND_{ICC} (8 bytes random)

$S = Dec(E_{IFD})$
verify M_{IFD}
verify RND_{ICC}
K_{ICC} (16 bytes random)
$K_{seed} = K_{IFD} \oplus K_{ICC}$
derive SK_{ENC} and SK_{MAC}
$SSC = RND_{IFD}$(lower half) ||
RND_{ICC} (lower half)
$R = RND_{ICC} || RND_{IFD} || K_{ICC}$
$E_{ICC} = Enc(R, K_{ENC})$
$M_{ICC} = MAC(E_{ICC}, K_{MAC})$

Threats to BAC

- **Symmetric BAC keys are derived from data contained in the optical MRZ (no infrastructure available):**
  - Passport serial number
  - Date of birth
  - Expiry date
- **Hence the keys are not generated from random key seed material.**
- **The structure of the MRZ information reduces the entropy of the key seed material.**
- **Data-base based attacks are possible.**
- **Obvious countermeasures:**
  - Use random serial number
  - Use alpha-numeric serialnumber

EAC

- **Asymmetric protocol to protect sensitive data (e.g.: DG3)**

- **Two components:**
  - Chip authentication (CA)
  - Terminal authentication (TA)

- **Uses either:**
  - EC Diffie Hellman (EC DH) key agreement or DH key agreement for CA
  - ECDSA or RSA based mechanisms for TA

- **MRTD public key used in CA is stored in DG14, authenticity is thus implicitly secured by LDS signature contained in EF.SOD. No “direct” certificate is issued for the CA public key**

- **On-chip asymmetric crypto computations needed**

EAC Chip Authentication

- **Chip Authentication is a protocol to authenticate the MRTD with respect to the terminal**
- **New session keys are derived from the common Diffie-Hellman / EC DH secret**
- **The session keys are used to restart the Secure Messaging between terminal and MRTD.**
- **The session keys generated by this protocol have a high entropy.**
- **The authentication of the chip is verified implicitly:**
  - The terminal concludes from the fact, that the MRTD is able to use the new SM session keys that the key material used in the DH protocol was authentic

EAC Terminal Authentication

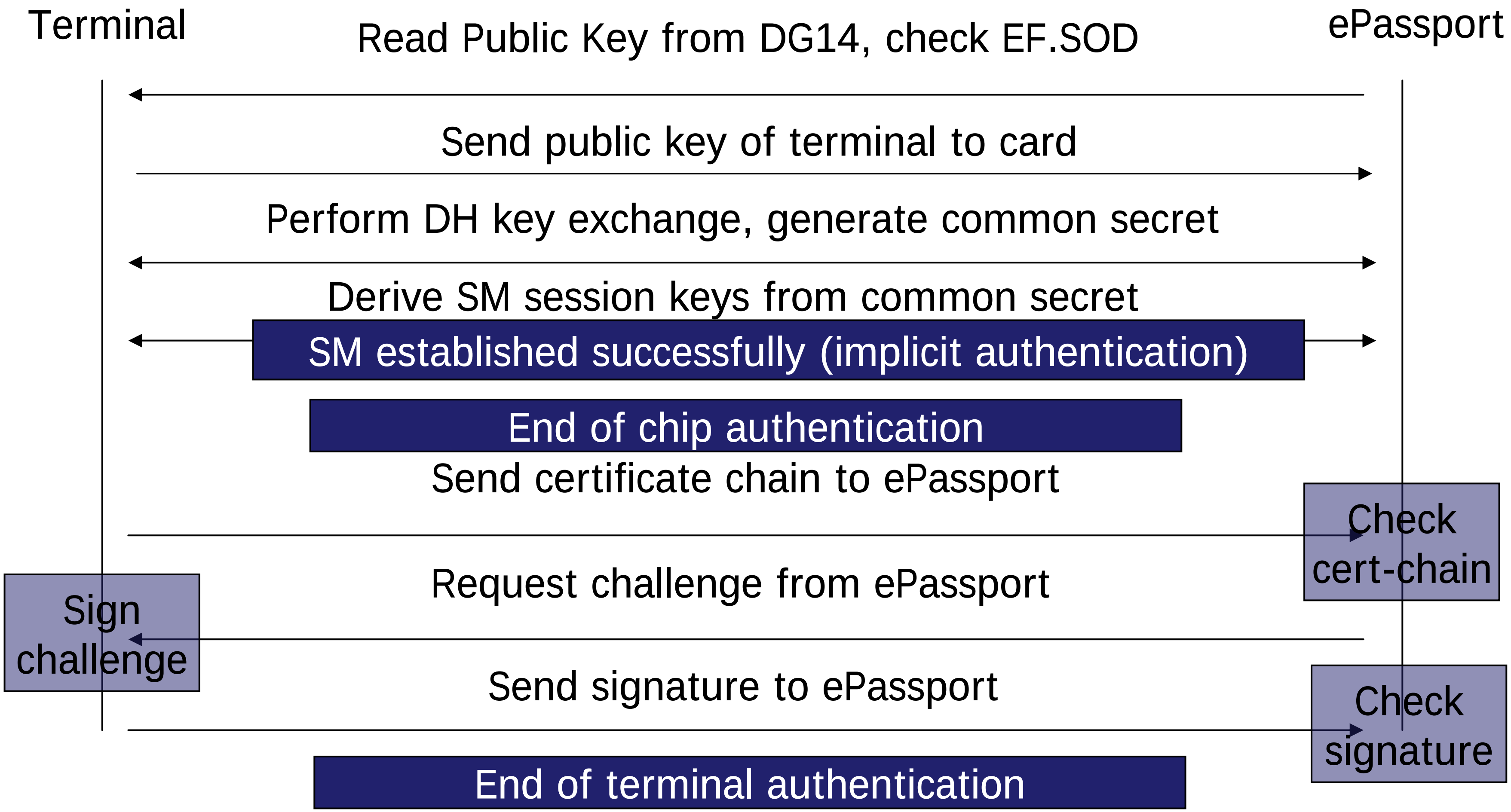
- **Terminal Authentication is a protocol to authenticate the terminal with respect to the MRTD**
- **The terminal must be in the possesion of an authentic key pair, that has to be authenticated (i.e. digitally signed) by a Certification Authority**
- **The MRTD accepts the authenticity of the terminal, if the Terminal Authentication is performed successfully**
- **Furthermore, the MRTD grants access to the sensitive data stored on the chip, if the Terminal Authentication is performed successfully**
  - Reading rights are encoded inside the certificate chain
- **Terminal Authentication implies that there is the need for a Public Key Infrastructure (PKI), that supplies the relevant certificates to the reader terminals.**

EAC Terminal Authentication

- **TA falls into two main parts:**
  - Transport of trusted public key of terminal to ICC
  - This is performed using a chain of card verifiable (CV) certificates

  - Proof, that the terminal is in posession of the private key corresponding to the public key transferred to the ICC
  - This is performed using an asymmetric challenge response mechanism in which :
    - The terminal signs a challenge received from the ICC
    - The ICC verifies this signature using the trusted public key of the terminal

EAC flow



Terminal

ePassport

Read Public Key from DG14, check EF.SOD

Send public key of terminal to card

Perform DH key exchange, generate common secret

Derive SM session keys from common secret

SM established successfully (implicit authentication)

End of chip authentication

Send certificate chain to ePassport

Check cert-chain

Request challenge from ePassport

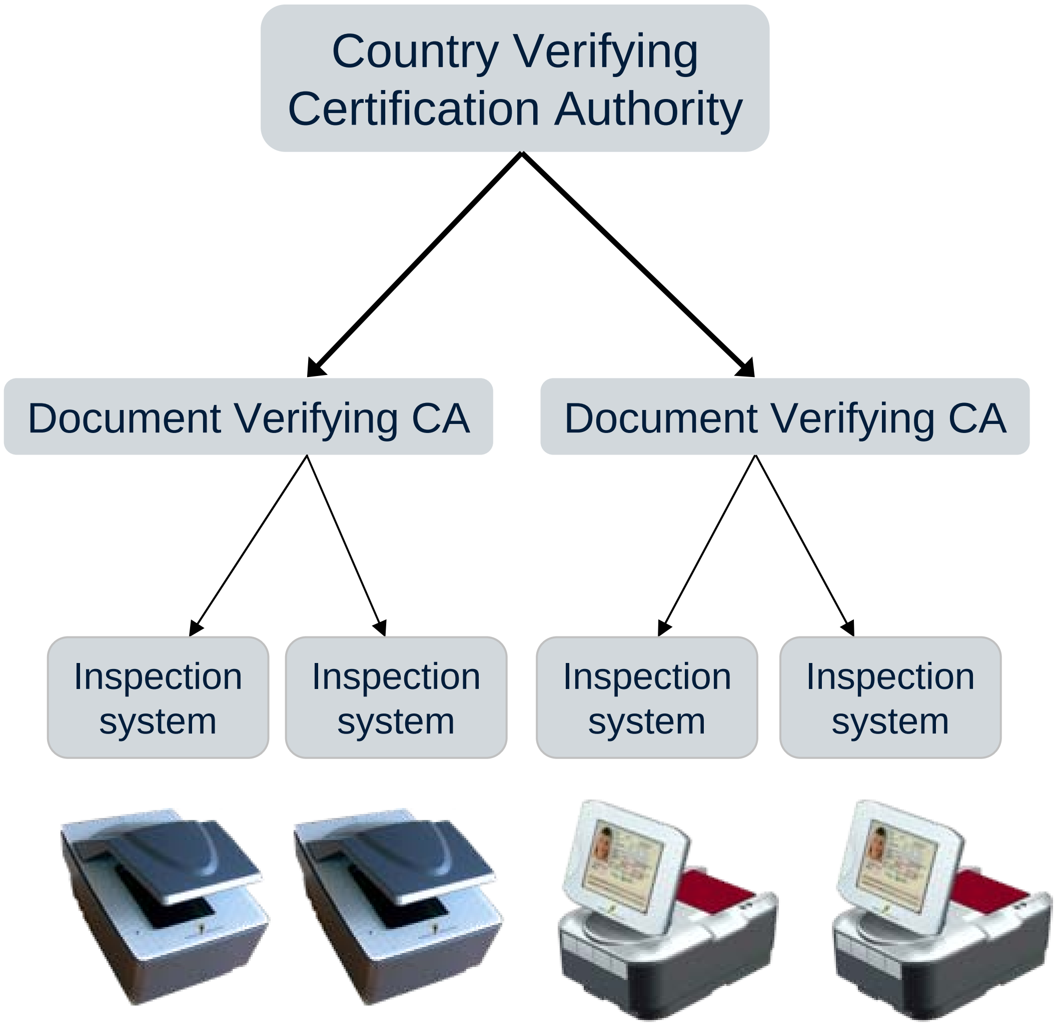
Sign challenge

Send signature to ePassport

Check signature

End of terminal authentication

EAC PKI



Country Verifying Certification Authority

Document Verifying CA

Document Verifying CA

Inspection system

Inspection system

Inspection system

Inspection system

Root of EAC PKI

- Issues Document Verifier certificates
- Typically a government task

DocumentVerifying CA (Organizational Units that use EAC terminals)

- Restricts validity period and access rights
- Issues Inspection System certificates

Terminals

Obtain access rights and validity periods via IS certificates.

Access rights are coded inside IS certificates, but can be restricted by both CVCA and CVCA. Only the acces right present in all three certificates is granted.

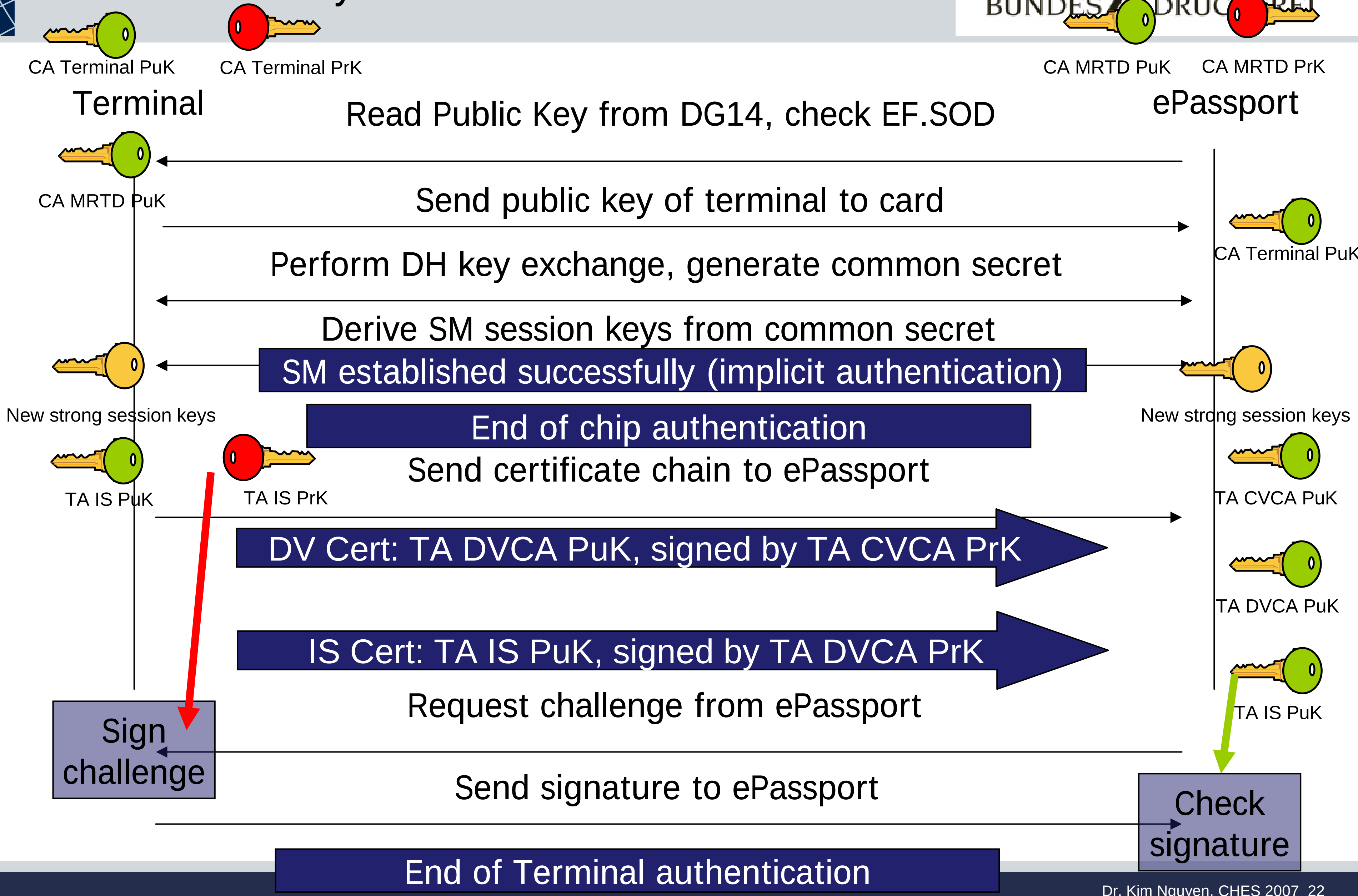
EAC Terminal certificate chain

- **Inspection system proves authenticity of its public key by means of a certificate chain.**

- **MRTD ICC contains root public key of its own CVCA.**
- **Terminal presents a chain of**
  - DV certificate (can be verified using root CVCA key)
  - IS certificate (can be verified using DV certificate)

- **In contrast to X.509 certificates, card verifiable certificates are used. These are not ASN.1 but TLV coded, so that the card can parse and interpret this objects**

EAC flow with key material



CA Terminal PuK

CA Terminal PrK

CA MRTD PuK

CA MRTD PrK

Terminal

ePassport

Read Public Key from DG14, check EF.SOD

CA MRTD PuK

Send public key of terminal to card

CA Terminal PuK

Perform DH key exchange, generate common secret

Derive SM session keys from common secret

SM established successfully (implicit authentication)

New strong session keys

New strong session keys

End of chip authentication

Send certificate chain to ePassport

TA IS PuK

TA IS PrK

TA CVCA PuK

DV Cert: TA DVCA PuK, signed by TA CVCA PrK

TA DVCA PuK

IS Cert: TA IS PuK, signed by TA DVCA PrK

TA IS PuK

Request challenge from ePassport

Sign challenge

Send signature to ePassport

Check signature

End of Terminal authentication

EAC Certificate validity

- **Problem: MRTD is not „online“, hence cannot connect to a „trusted clock“**
- **Hence offline mechanism must be used.**

- **MRTD has its own internal data (initially date of production)**

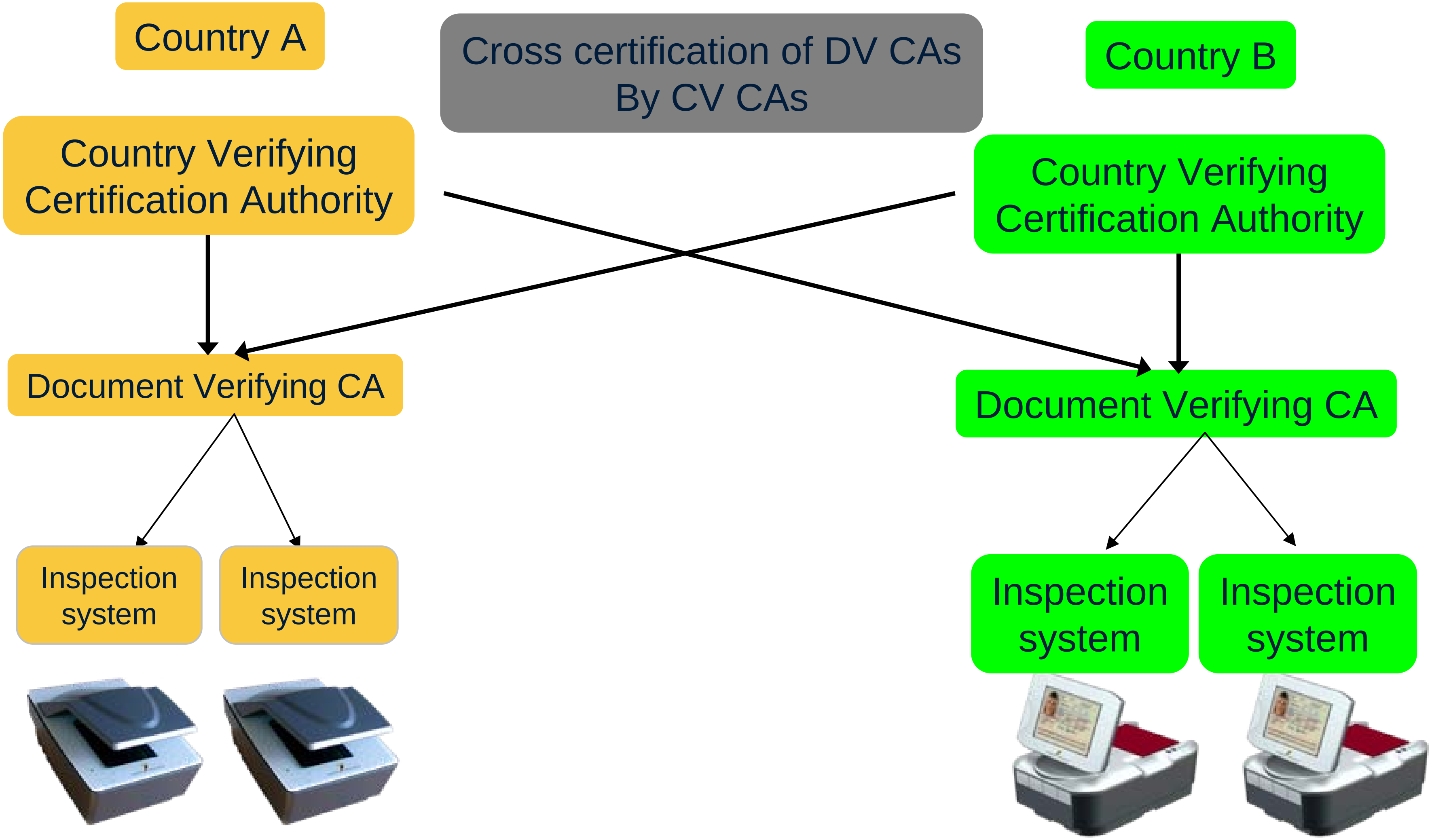
- **If MRTD verifies terminal certificates successfully, it compares its internal time with the issuing date of the certificate and updates the internal time if necessary and if the certificate was issued by a trusted CA (some more details omitted here)**

- **External certificates are reected if their expiry date lies before the MRTDs internal date.**

EAC root CVCA public key

- **MRTD contains root public key of its own national CVCA**
- **The root public key accepted by the MRTD as trusted is displayed inside the MRTD ICAO application.**
- **The root public key can be updated using link certificates.**
  - The MRTD verifies the validity of the link certificate against the trusted root public key.
  - In case of successfull verification the MRTD imports the new trusted CVCA root public key
  - It updates the information on trusted root keys accordingly.

EAC PKI - Crosscertificates



Country A

Cross certification of DV CAs By CV CAs

Country B

Country Verifying Certification Authority

Country Verifying Certification Authority

Document Verifying CA

Document Verifying CA

Inspection system

Inspection system

Inspection system

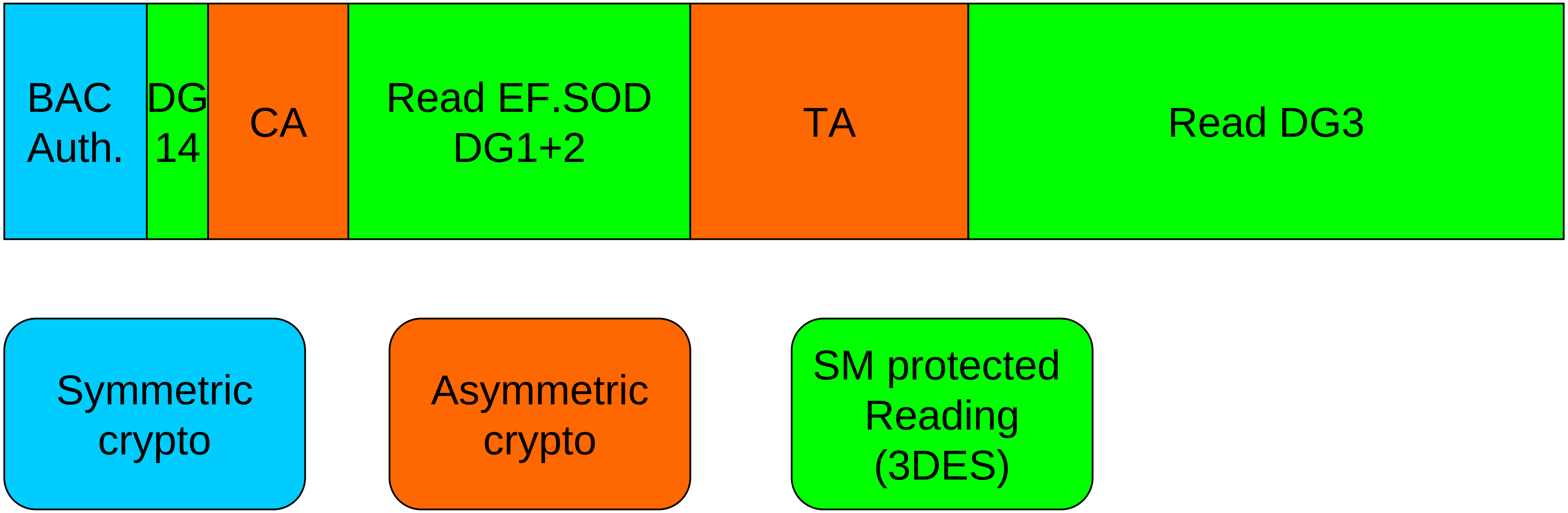
Inspection system

EAC – Implementation Issues

- **Kryptographic primitives used:**
  - Symmetric 3DES based encryption and macing (Retail-MAC)
  - Hashfunction computation (SHA-1/SHA-2)
  - Asymmetric DH based key exchange
  - Asymmetric signature computation/verification
- **Security of implementation**
  - Established on level CC EAL4+ according to Protection Profile „MRTD with EAC application“
- **Specific tasks of microcontroller in MRTD**
  - BAC based symmetric authentication with session key derivation
  - DH key exchange with session key derivation
  - Signature verification
  - Secure Messaging (based on 3DES)

EAC – Implementation Issues

- **EAC readout process for ECC based authenticaton (approximately to scale):**



BAC Auth.

DG 14

CA

Read EF.SOD DG1+2

TA

Read DG3

Symmetric crypto

Asymmetric crypto

SM protected Reading (3DES)

Summary

- **MRTDs are proving to be a new innovative area for the integration of symmetric and asymmetric cryptographic mechanisms and protocols**
- **2nd generation MRTDs as implemented in the EU right now make use of strong asymmetric primitive as a mandatory deature**
- **The usage of contactless security controllers leads to special requirements both for hardware as well as authentication protocols implemented.**

Thank you for your attention!

Certainly, there are questions?