

Contactless Authentication protocols for MRTDs: BAC & EAC

Dr. Kim Nguyen

Bundesdruckerei GmbH

Rfid Security 2006, Graz, 2006-07-13



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

- 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 separate 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)

- **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.

- **Content of 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

- **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

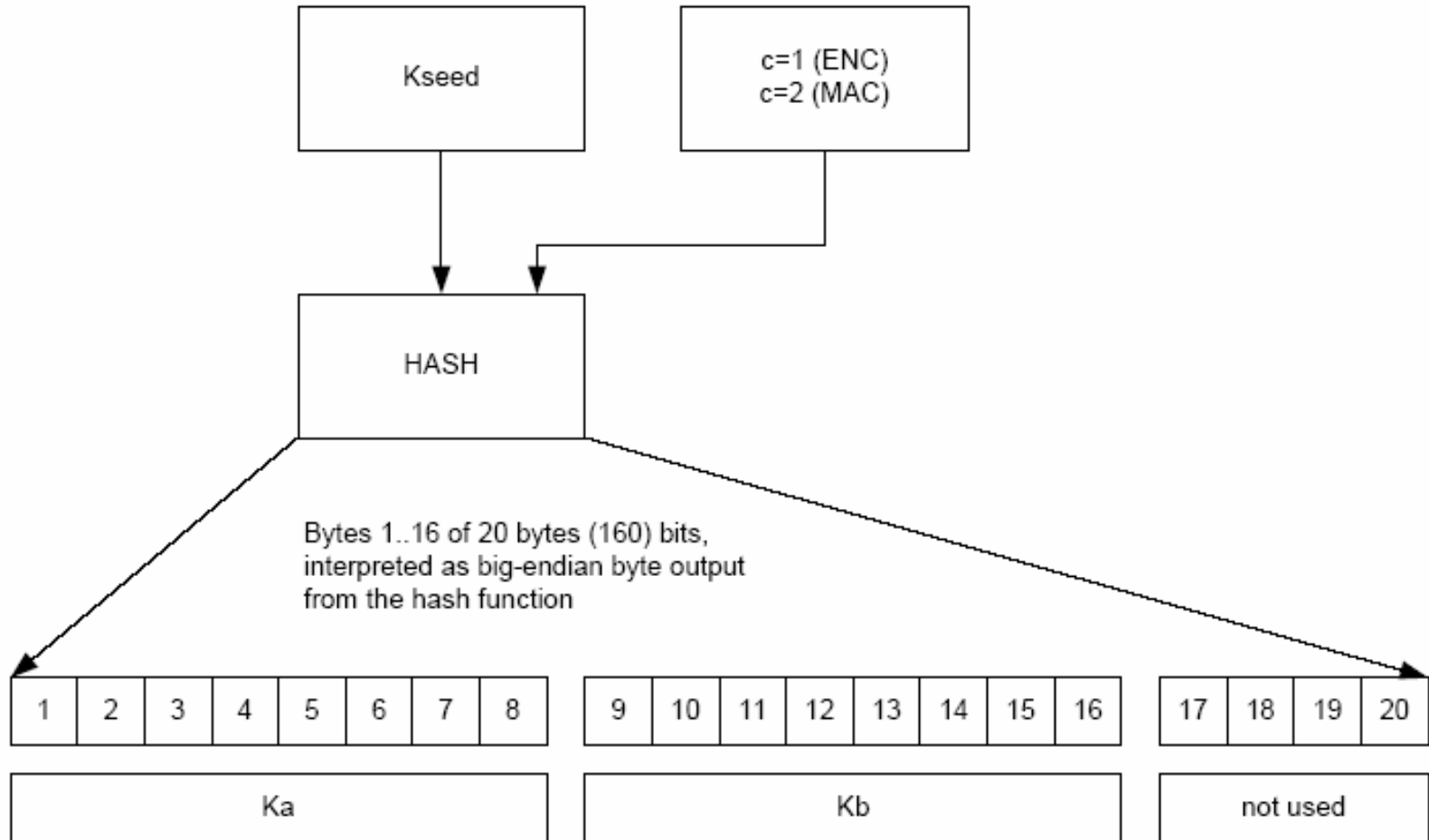
- **Two types of authentication protocols are implemented for MRTDs:**
 - ◆ Authentication protocol without need for infrastructure
 - Symmetric (3DES based) protocol
 - Protocol initialisation only requires information that is optically available on the passport
 - => Basic Access Control (BAC)
 - ◆ Authentication protocols with need for infrastructure
 - Asymmetric protocol
 - Protocol requires access to certificates for terminals (PKI)
- **Furthermore we have passive authentication (verification of LDS signature)**
 - ◆ This requires a PKI in order to have the Country Signing and Document signer certificates available.

- **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)**

- 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} :
 - ◆ $c = \text{'0x 00 00 00 01'}$ for K_{ENC}
 - ◆ $c = \text{'0x 00 00 00 02'}$ for K_{MAC}



■ Notations:

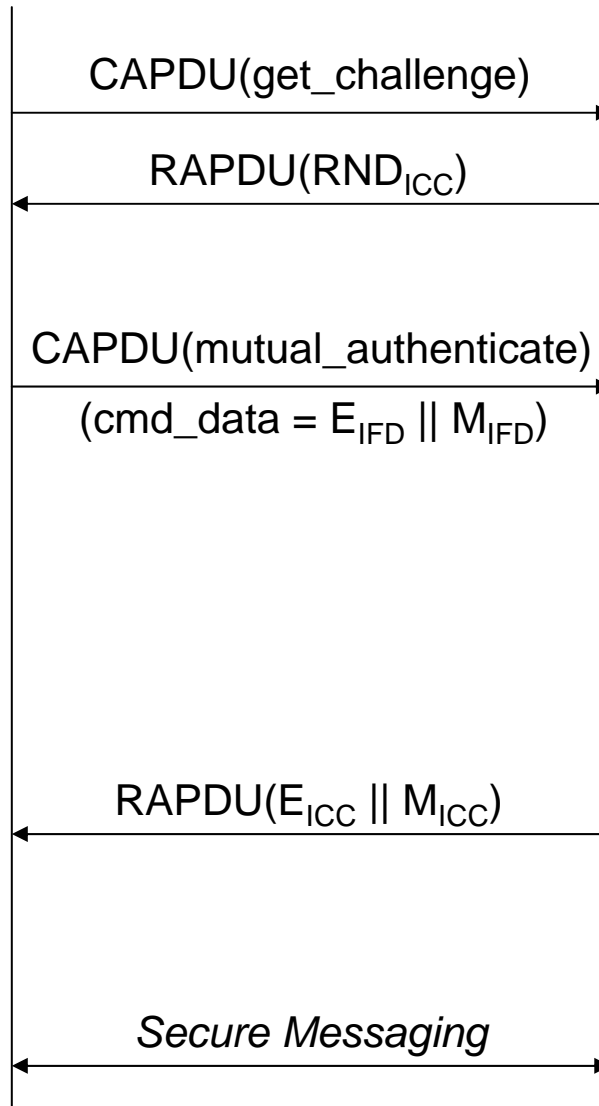
- ◆ IFD denotes the Interface Device (Reader Terminal)
- ◆ ICC denotes the chip inside the MRTD
- ◆ CAPDU denotes a command sent from the IFD to the ICC
- ◆ RAPDU denotes a response sent from the ICC to the IFD
- ◆ $K_{\text{IFD/ICC}}$ denotes a random value chosen by the IFD/ICC
- ◆ $\text{Enc}(S,K)$ denotes a 3DES encryption of data S using key K
- ◆ $\text{Dec}(S,K)$ denotes a 3DES decryption of cryptogram S using key K
- ◆ $\text{MAC}(S,K)$ denotes a 3DES Retail MAC of data S using key K
- ◆ SK_{ENC} and SK_{MAC} denotes session keys derived by IFD and ICC
- ◆ SSC denotes the sent sequence counter derived by IFD and ICC

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}
 $SSC = RND_{IFD}(\text{lower half}) ||$
 $RND_{ICC}(\text{lower half})$



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}
 $SSC = RND_{IFD}(\text{lower half}) ||$
 $RND_{ICC}(\text{lower half})$
 $R = RND_{ICC} || RND_{IFD} || K_{ICC}$
 $E_{ICC} = Enc(R, K_{ENC})$
 $M_{ICC} = MAC(E_{ICC}, K_{MAC})$

- **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

- **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
- **Chip public key used in CA is stored in DG14, authenticity is thus secured by LDS signature contained in EF.SOD**
- **On-chip asymmetric crypto computations needed**

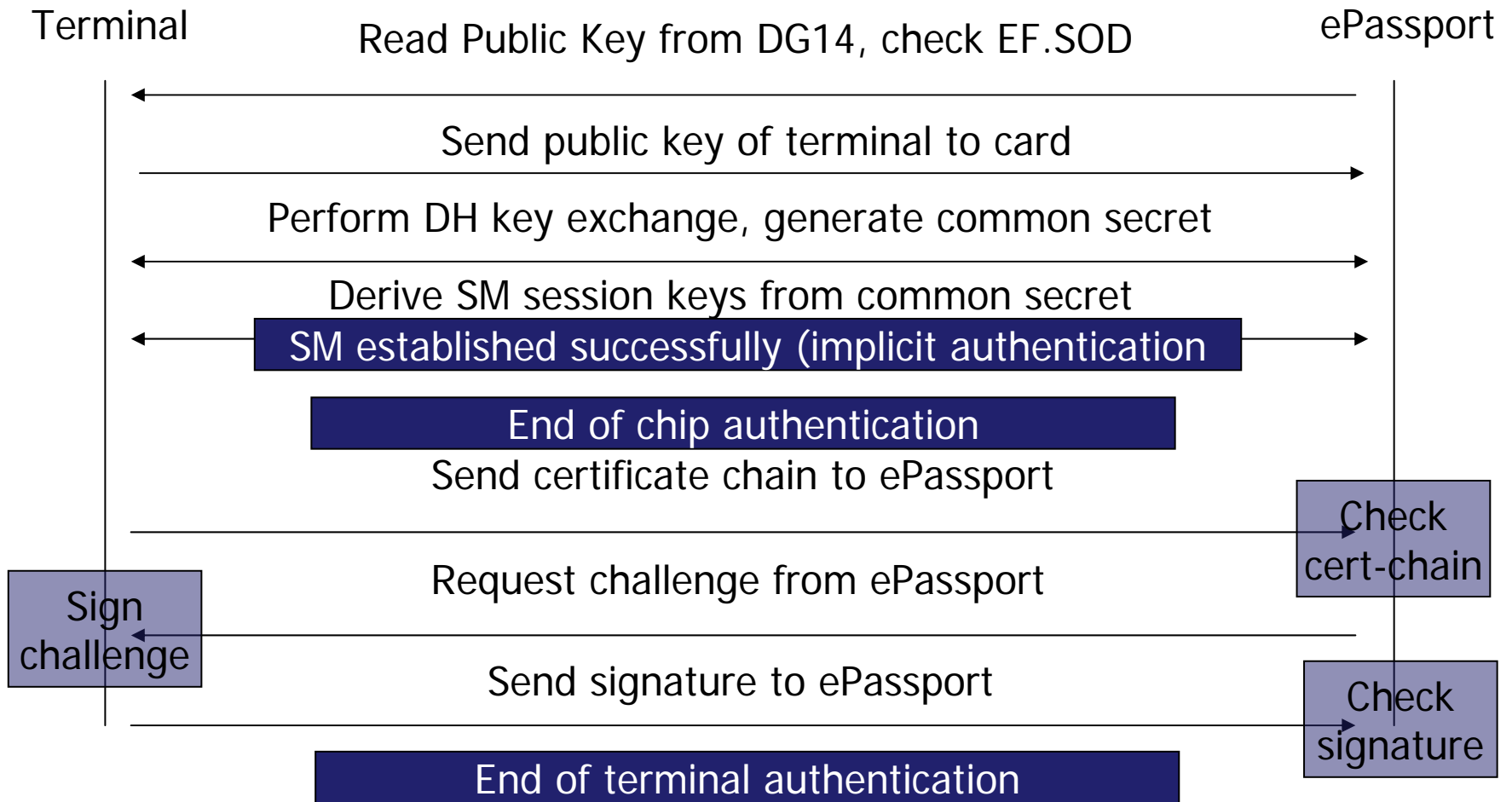
- **Chip Authentication is a protocol to authenticate the MRTD with respect to the terminal**
- **New session keys are derived from the common Diffie-Hellman / EC DH secret**
- **The session keys are used to restart the Secure Messaging between terminal and MRTD.**
- **The session keys generated by this 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

- **Terminal Authentication is a protocol to authenticate the terminal with respect to the MRTD**
- **The terminal must be in the possession 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.**

■ 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 possession 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



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

Document Verifying 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 DVCA. Only the access right present in all three certificates is granted.

- **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



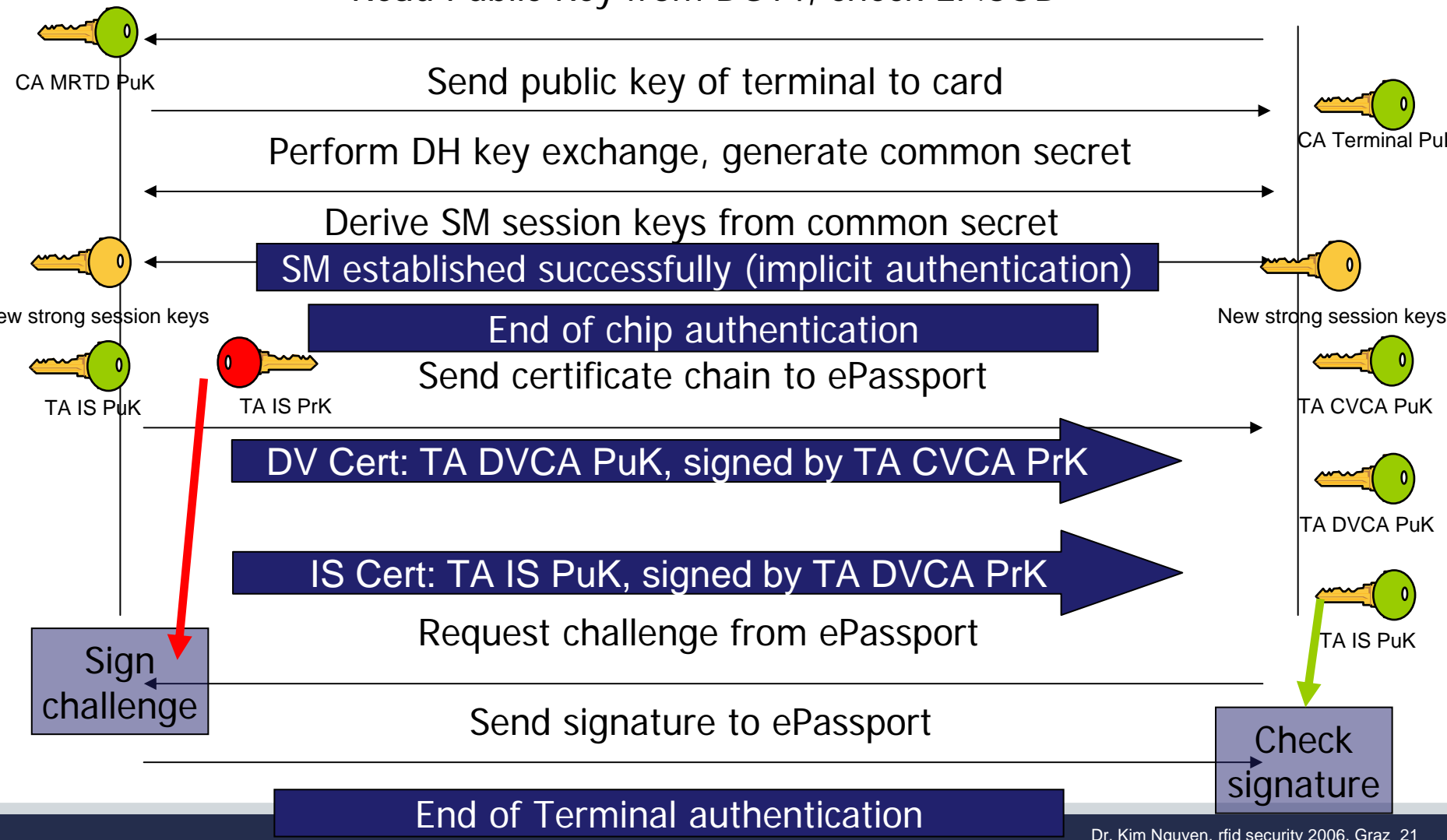
CA Terminal PuK CA Terminal PrK

CA MRTD PuK CA MRTD PrK

Terminal

ePassport

Read Public Key from DG14, check EF.SOD

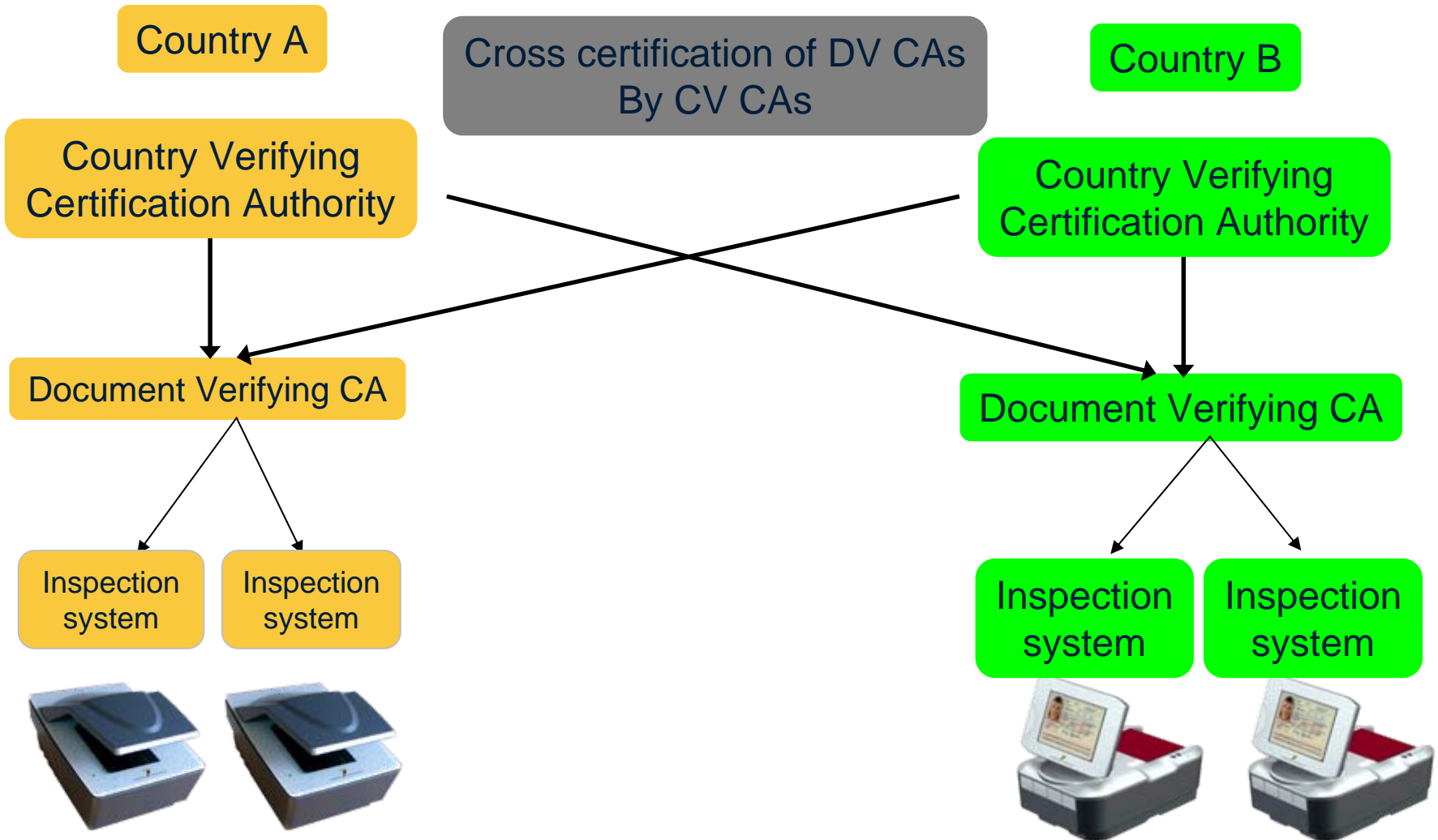


- **Problem: MRTD is not „online“, hence cannot connect to a „trusted clock“**
- **Hence offline mechanism must be used.**
- **MRTD has its own internal data (initially date of production)**
- **If MRTD verifies terminal certificates successfully, it compares its internal time with the issuing date of 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 rejected if their expiry date lies before the MRTDs internal date.**

- **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 successful verification the MRTD imports the new trusted CVCA root public key
 - ◆ It updates the information on trusted root keys accordingly.



- **Demonstration of prototype EAC implementation**

Thank you for your attention!

Certainly, there are questions?