**Security and Cryptography 2015**

**Mirosław Kutyłowski**

**grading criteria:** 50% exam, 50% assignments

**skills to be learned:** developing end-to-end security systems, they must be flawless!

**rules:** do not memorize the standards, they come and go. Only the skills are important

**exam date:** TBA (quickly to enable internships in February)

---

## I. Example to learn from: PKI failure

reasons for PKI failure (According to Schneier):

1. whom we trust and for what? why CA should be trusted??
2. who is using my key? (private key - there are really no clones??)
3. how secure is the verifying computer? (no cryptography can help is the verifier software is cheating)
4. who is the signer? (ambiguity unless there is a trustworthy ID registry)
5. is CA an authority? (really not an authority for data contained in the certificate. Certificate bsed on fake documents...)
6. is the user part of the security design? (no, the user is free to behave in a stupid way)
7. separation CA and RA brings new threats
8. How did CA verify the certificate holder? (certificate issued for ..., but how to know that this was really this person)
9. How secure are the certification practices? (revocation, etc)
10. the customers wish to run single-sign-on

**Answers:** yes, but

- tradition in Nordic countries
- honest system participants
- the best one can do
- ...

  no solutions from crypto community

**MAJOR PROBLEM:** how to design/buy sound systems?

## II. COMMON CRITERIA FRAMEWORK

http://www.commoncriteriaportal.org/

**Idea:**

- standardize the process of
  - designing products (Security Target ST),
  - designing requirements (Protection Profile, PP)
  - evaluation of products (licensed labs checking conformance of implementation with the documentation)
- international agreement of bodies from some countries (USA, France, UK, Germany, India, Turkey, Sweden, Spain, Australia, Canada, Malaysia, Netherlands, Korea, New Zeland, Italy, Turkey) but Israel only "consuming", no Poland, China, Singapore,
- idea: ease the process,
- support for certification industry

**Value:**

- CC certification does not mean a product is secure
- it only says that is has been developed according to PP
- assurance level concerns only the stated requirements , e.g. trivial requirements ⇒ high EAL level (common mistake in public procurement: EAL level ... without specifying PP
- clean up the zoo of different assumptions, descriptions, ...

**Example for PP: BAC (Basic Access Control)**

- encryption primitive $\mathrm{EM}(K, S) = \mathrm{Enc}(\mathrm{KB}_{\mathrm{Enc}}, S) \| \mathrm{MAC}(\mathrm{KB}_{\mathrm{Mac}}, \mathrm{Enc}(\mathrm{KB}_{\mathrm{Enc}}, S), S)$ where $K = \{\mathrm{KB}_{\mathrm{Enc}}, \mathrm{KB}_{\mathrm{Mac}}\}$
- steps:
  1. The MRTD chip sends the nonce $r_{\mathrm{PI}\mathbb{C}C}$ to the terminal
  2. The terminal sends the encrypted challenge $e_{\mathrm{PCD}} = \mathrm{EM}(K, r_{\mathrm{PCD}}, r_{\mathrm{PI}\mathbb{C}C}, K_{\mathrm{PCD}})$ to the MRTD chip, where $r_{\mathrm{PI}\mathbb{C}C}$is the MRTD chip's nonce, $r_{\mathrm{PCD}}$ is the terminal's randomly chosen nonce, and $K_{\mathrm{PCD}}$ is keying material for the generation of the session keys.
  3. The MRTD chip decrypts and verifies $r_{\mathrm{PI}\mathbb{C}C}$, responds with $e_{\mathrm{PICC}} = \mathrm{EM}(K, r_{\mathrm{PICC}}, r_{\mathrm{PCD}}, K_{\mathrm{PICC}})$
  4. The terminal decrypts and verifies $r_{\mathrm{PCD}}$

5. both sides derive $K_{\text{Enc}}$, $K_{\text{Mac}}$ from master key $K_{\text{PICC}}$ XOR $K_{\text{PCD}}$ and sequence number derived from randoms (key derivation function)

- $K$ derived from information available on the machine readable zone (optical)
- implementation: biometric passports.
- simple system. Really?

**Common Criteria Protection Profile Machine Readable Travel Document with ICAO Application, Basic Access Control BSI-CC-PP-0055**

## 1. Introduction

### 1.1 PP reference

1 Title: Protection Profile - Machine Readable Travel Document with ICAO Application and Basic Access Control (MRTD-PP)

Sponsor: Bundesamt für Sicherheit in der Informationstechnik CC Version: 3.1 (Revision 2)

Assurance Level: The minimum assurance level for this PP is EAL4 augmented.

General Status: Final

Version Number: 1.10

Registration: BSI-CC-PP-0055

Keywords: ICAO, machine readable travel document, basic access control

### 1.2 TOE Overview

- Target of Evaluation
- "is aimed at potential consumers who are looking through lists of evaluated TOEs/Products to find TOEs that may meet their security needs, and are supported by their hardware, software and firmware"
- important sections:
  - Usage and major security features of the TOE
  - TOE type
  - Required non-TOE hardware/software/firmware
- Definition, Type

  which parts, which general purpose, which functionalities are present and which are missing, eg. ATM card with no contactless payments

- Usage and security features

  crucial properties of the system (high level) and security features from the point of view of the security effect and not how it is achieved

- life cycle

  the product in the whole life cycle including manufacturer and destroying

- Required non-TOE hardware/software/firmware: other components that can be crucial for evaluation

## 2. Conformance Claim

- CC Conformance Claim: version of CC
- PP claim: other PP taken into account in a plug-and-play way
- Package claim: which EAL package level

**EAL packages:**

- The CC formalizes assurance into 6 categories (the so-called "assurance classes" which are further subdivided into 27 sub-categories (the so-called "assurance families"). In each assurance family, the CC allows grading of an evaluation with respect to that assurance family.
- assurance classes:
  - → development:
    - ADV_ARC - 1 1 1 1 1 1 architecture requirements
    - ADV_FSP 1 2 3 4 5 5 6 functional specifications
    - ADV_IMP - - - 1 1 2 2 implementation representation
    - ADV_INT - - - - 2 3 3 “is designed and structured such that the likelihood of flaws is reduced and that maintenance can be more readily performed without the introduction of flaws”?
    - ADV_SPM - - - - - 1 1 security policy modeling
    - ADV_TDS - 1 2 3 4 5 6 TOE design
  - → guidance documents
    - AGD_OPE 1 1 1 1 1 1 1 Operational user guid ance
    - AGD_PRE 1 1 1 1 1 1 1 Preparative procedures
  - → life-cycle support
    - ALC_CMC 1 2 3 4 4 5 5 CM capabilities
    - ALC_CMS 1 2 3 4 5 5 5 CM scope
    - ALC_DEL - 1 1 1 1 1 1 Delivery

        - ALC_DVS - - 1 1 1 2 2 Development securit
        - ALC_FLR - - - - - - - Flaw remediation
        - ALC_LCD - - 1 1 1 1 2 Life-cycle definition
        - ALC_TAT - - - 1 2 3 3 Tools and techniques
    - security target evaluation
        - ASE_CCL 1 1 1 1 1 1 1 Conformance claims
        - ASE_ECD 1 1 1 1 1 1 1 Extended components definition
        - ASE_INT 1 1 1 1 1 1 1 ST introduction
        - ASE_OBJ 1 2 2 2 2 2 2 Security objectives
        - ASE_REQ 1 2 2 2 2 2 2 Security requirements
        - ASE_SPD - 1 1 1 1 1 1 Security problem definition
        - ASE_TSS - 1 1 1 1 1 1 TOE summary specification
    - tests
        - ATE_COV 1 2 2 2 3 3 Coverage
        - ATE_DPT 1 1 3 3 4 Depth
        - ATE_FUN 1 1 1 1 2 2 Functional tests
        - ATE_IND 1 2 2 2 2 2 3 Independent testing
    - vulnerability assesment
        - AVA_VAN 1 2 2 3 4 5 5 Vulnerability analysis

- for example, a product could score in the assurance family developer test coverage (ATE_COV):
    - 0: It is not known whether the developer has performed tests on the product;
    - 1: The developer has performed some tests on some interfaces of the product;
    - 2: The developer has performed some tests on all interfaces of the product;
    - 3: The developer has performed a very large amount of tests on all interfaces of the product
- example more formal: ALC_FLR
    - ALC_FLR.1:
        - The flaw remediation procedures documentation shall describe the procedures used to track all reported s ecurity flaws in each release of the TOE.

    - The flaw remediation procedures sha ll require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

    - The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

    - The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

  - ALC_FLR.2:

    - first four like before

    - The flaw remediation procedures sh all describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.

    - The procedures for processing reported security flaws shall ensure that any reported flaws are remediated and the remediation procedures issued to TOE users.

    - The procedures for processing repor ted security flaws shall provide safeguards that any corr ections to these security flaws do not introduce any new flaws.

    - The flaw remediation guidance sha ll describe a means by which TOE users report to the developer any susp ected security flaws in the TOE.

  - ALC_FLR.3:

    - first 5 as before

    - The flaw remediation procedures shall include a procedure requiring timely response and the automatic distri bution of security flaw reports and the associated corrections to registered users who might be affected by the security flaw.

    - next 3 as before

    - The flaw remediation guidance shall describe a means by which TOE users may register with the developer, to be eligible to receive security flaw reports and corrections.

    - The flaw remediation guidance shall iden tify the specific points of contact for all reports and enquiries about security issues involving the TOE.

- 7 predefined ratings, called evaluation assurance levels or EALs. called EAL1 to EAL7, with EAL1 the lowest and EAL7 the highest

- Each EAL can be seen as a set of 27 numbers, one for each assurance family. EAL1 assigns a rating of 1 to 13 of the assurance families, and 0 to the other 14 assurance families, while EAL2 assigns the rating 2 to 7 assurance families, the rating 1 to 11 assurance families, and 0 to the other 9 assurance families

- monotonic: EALn+1 gives at least the same assurance level as EALn in each assurance families

- levels:
    - EAL1: Functionally Tested:
        - correct operation, no serious threats
        - minimal effort from the manufacturer
    - EAL2: Structurally Tested
        - delivery of design information and test results,
        - effort on the part of the developer than is consistent with good commercial practice.
    - EAL3: Methodically Tested and Checked
        - maximum assurance from positive security engineering at the design stage without substantial alteration of existing sound development practices.
        - developers or users require a moderate level of independently assured security, and require a thorough investigation of the TOE and its development without substantial re-engineering.
    - EAL4: Methodically Designed, Tested and Reviewed
        - maximum assurance from positive security engineering based on good commercial development practices which, though rigorous, do not require substantial specialist knowledge, skills, and other resources.
        - the highest level at which it is likely to be economically feasible to retrofit to an existing product line.
    - EAL5: Semiformally Designed and Tested
    - EAL6: Semiformally Verified Design and Tested
    - EAL7: Formally Verified Design and Tested

**CEM -Common Evaluation Methodology**

- given CC documentation, EAL classification etc, perform a check
- idea: evaluation by non-experts, semi-automated, mainly paper work
- mapping:
    - assurance class ⇒ activity
    - assurance component ⇒ sub-activity
    - evaluator action element ⇒ action
    - developer action element ⇒ work-unit

- content and presentation of evidence element ⇒ work unit

- responsibilities:
  - sponsor: requesting and supporting an evaluation. different agreements for the evaluation (e.g. commissioning the evaluation), providing evaluation evidence.
  - developer: produces TOE, providing the evidence required for the evaluation on behalf of the sponsor.
  - evaluator: performs the evaluation tasks required in the context of an evaluation, performs the evaluation sub-activities and provides the results of the evaluation assessment to the evaluation authority.
  - evaluation authority: establishes and maintains the scheme, monitors the evaluation conducted by the evaluator, issues certification/validation reports as well as certificates based on the evaluation results
- verdicts: pass, fail, inconclusive
- parts:
  - evaluation input task (are all documents available to perform evaluation?)
  - evaluation sub-activities
  - evaluation output task (de scribe the Observation Report (OR) and the Evaluation Technical Report (ETR )).
  - demonstration of the technical competence task

## 3 Security Problem Definition

- **Object Security Problem (OSP)**: "The security problem definition defines the security problem that is to be addressed.

  – axiomatic. deriving the security problem definition outside the CC scope

  – the usefulness of the results of an evaluation strongly depends on the security problem definition.

  – spend significant resources and use well-defined processes and analyses to derive a good security problem definition.

- good example:

  Secure signature-creation devices must, by appropriate technical and operational means, ensure at the least that:

  1) The signature-creation-data used for signature-creation can practically occur only once, and that their secrecy is reasonably assured;

  2) The signature-creation-data used for signature-creation cannot, with reasonable assurance, be derived and the signature is protected against forgery using currently available technology;

3) The signature-creation-data used for signature-creation can be reliably protected by the legitimate signatory against the use of others

- **assets:** entities that someone places value upon. Examples of assets include: - contents of a file or a server; - the authenticity of votes cast in an election; - the availability of an electronic commerce process; - the ability to use an expensive printer; - access to a classified facility.

  no threat no asset

- **Threats:** threats to assets

- **Assumptions:** assumptions are acceptable, where certain properties of the TOE environment are already known,

  – but not when they are derived from specific properties of the TOE

- **Security objectives:**

  - "The security objectives are a concise and abstract statement of the intended solution to the problem defined by the security problem definition. Their role:

    - a high-level, natural language solution of the problem;

    - divide this solution into partwise solutions, each addressing a part of the problem;

    - demonstrate that these partwise solutions form a complete solution to the problem.

  - bridge between the security problem and Security Functional Requirements (SFR)

- **mapping objectives to threats**: table, each threat shoud be covered, each objective has to respond to some threat

  answers to questions:

  - what is really needed?

  - have we forgot about something?

- **rationale:** verifiable explanation why the mapping is sound

4. **SFR (Security Functional requirements)**

   - *The SFRs are a translation of the security objectives for the TOE. They are usually at a more detailed level of abstraction, but they have to be a complete translation (the security objectives must be completely addressed) and be independent of any specific technical solution (implementation). The CC requires this translation into a standardised language for several reasons: - to provide an exact description of what is to be evaluated. As security objectives for the TOE are usually formulated in natural language, translation into a standardised language enforces a more exact description of the functionality of the TOE. - to allow comparison between two STs. As different ST authors may use different terminology in describing their security objectives, the standardised language enforces using the same terminology and concepts. This allows easy comparison.*

   - predefined classes:

     - Logging and audit class FAU

     - Identification and authentication class FIA

- Cryptographic operation class FCS

- Access control families FDP_ACC, FDP_ACF

- Information flow control families FDP_IFC, FDP_IFF

- Management functions class FMT

- (Technical) protection of user data families FDP_RIP, FDP_ITT, FDP_ROL

- (Technical) protection of TSF data class FPT

- Protection of (user) data during communication with external entities families FDP_ETC, FDP_ITC, FDP_UCT, FDP_UIT, FDP_DAU, classes FCO and FTP

- customizing SFRs: refinement (more requirements), selection (options), assignment (values), iterations (the same component may appear at different places with different roles)

- rules:

  check dependencies between SFR

  security objectives must follow from SFR's

  if possible, use only standard SFR's