The Common Criteria

John Franco

*Electrical Engineering and Computing Systems*
**History of Security Evaluation**

**The Orange Book (1983)**

Basic requirements for assessing effectiveness of security controls

Used to evaluate, classify, select computer systems for processing, storage, retrieval of classified info

Centerpiece of the “Rainbow Series”

**Policy:** explicit, well-defined

**Accountability:** enforcement a must

**Assurance:** on each component
  - Operational assurance: architecture, integrity, recovery
  - Life-cycle assurance: design verification, safe distribution

**Continuous protection assurance**

**Four levels of evaluation:**
  - minimal, discretionary, mandatory, verified
History of Security Evaluation


Procedures for evaluating computer systems for procurement

Considered (selected list):
- Password management
- Audits in trusted systems
- Access control in trusted systems
- Configuration management
- Design documentation development
- Trusted distribution
- Security modeling
- Formal verification of systems
- Trusted facility management
- Object reuse in trusted systems
- Trusted recovery
- Security testing and testing documentation
- Procurement: how to evaluate a bidder's proposal
- Information System Security Officer responsibilities
- Covert channel analysis
History of Security Evaluation

**Information Technology Security Evaluation Criteria (1990)**

Criteria for evaluating computer products and systems for security

Developed by France, Germany, the Netherlands, United Kingdom

Objectives: Confidentiality, Availability, Integrity

Evaluation and certification processes are defined

Seven evaluation levels: ↑ confidence in meeting security target

- **E0**: Inadequate – does not meet security target
- **E1**: Informal description of architecture. Functional testing satisfies the security target
- **E2**: E1 + informal description of detailed design + configuration control + approved distribution procedure
- **E3**: E2 + hardware, source code details
- **E4**: E3 + formal model of security policy supporting security target
- **E5**: E4 + correspondence between detailed design and source code and hardware drawings
- **E6**: E5 + security enforcing functions and architecture is specified formally, consistent with formal mode of security policy
Examples of formal models of security policy
Bell-La Padula model (US DoD, Mitre Corp, 1976) for MLS
state transition model of computer security policy
describes access control rules with security labels on objects
and clearances (top secret to unclassified) for subjects
Objective: data confidentiality

Subject: active entity such as a process
Object: passive entity such as a file

One of the axioms, called the $*$-property, prohibits a subject from simultaneously having read access to one object at a given security level and write access to another object at a lower security level. Its purpose is to prevent subjects from moving data of a given security level to an object marked with a lower security level.
History of Security Evaluation


Examples of formal models of security policy

Biba Model (US DoD, Mitre Corp – 1977)
- state transition model of computer security policy
- describes access control rules designed to ensure data integrity
  - Data and subjects placed in levels
  - Subjects do not corrupt data in a higher level

Clark-Wilson model (US, academic – 1987)
- well-formed transactions (ops transitioning from consistent state to consistent state) enforce and certify a system integrity policy.
  - More useful in industrial, rather than military systems

The Eizenberg model (Germany - 1989)
- Access control rights that vary with time

The Landwehr model (US Naval Research Labs – 1984)
- A security model should be derived from a specific application
  - Intended for use in MLS
- Corrects defect in the Bell-La Padula model
The Common Criteria

- International standard for computer security certification created to pretty much replace TCSEC and ITSEC for a uniform standard internationally.

- Framework for users to specify security functional and assurance requirements (SFRs and SARs respectively) through Protection Profiles (PPs) designed to absorb and progressively weaken attack.

- Vendors make claims against these specs and independent labs or the NSA can determine whether claims are valid.
The Common Criteria

Process:

<table>
<thead>
<tr>
<th>Security Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laws, organizational security policies, etc, which define the context in which the TOE is to be used. Threats present in the environment are also included.</td>
</tr>
</tbody>
</table>

| TOE - Target of Evaluation | An Information Technology (IT) product or system and its associated administrator and user guidance documentation that is the subject of an evaluation |

Needed:

**TSP**: ToE Security Policy - rules that regulate how assets are managed, protected, and distributed within a ToE

**TSF**: Security Function – implementation of a security policy as well as a security objective whereby the policy is 'enforced'. Security functions are designed to withstand threats, risks and vulnerabilities.

**TSFI**: ToE Security Function Interface so users can gain access To the ToE.
The Common Criteria

Process:

<table>
<thead>
<tr>
<th>Security Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>A statement of intent to counter the identified threats and/or satisfy intended organizational security policies and assumptions.</td>
</tr>
</tbody>
</table>

| ST - Security Target | Set of security requirements and specification to be used as the basis for evaluation of an identified TOE. The ST may claim conformance to one or more Protection Profiles (PPs) and forms the basis of the evaluation. |

PPs: detailed level of security requirements and standards pertinent to specific technology or security risk area based on the overall CC framework or IT product or technology

Security requirements that can be certified as complete, consistent and technically sound in addressing threats that exist in a specified environment
The Common Criteria Process:

<table>
<thead>
<tr>
<th>TOE Security Requirements</th>
<th>The refinement of the IT security objectives into a set of technical requirements for security functions and assurance, covering the TOE and its IT environment.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TSP</strong> – TOE Security Policy</td>
<td>A set of rules that regulate how assets are managed, protected, and distributed within a TOE.</td>
</tr>
<tr>
<td><strong>SF</strong> – Security Function</td>
<td>A part or parts of the TOE that have to be relied upon for enforcing a closely related subset of the rules from the TSP.</td>
</tr>
<tr>
<td><strong>SFP</strong> – Security Function Policy</td>
<td>The security policy enforced by a SF.</td>
</tr>
</tbody>
</table>
The Common Criteria

Process:

**Classes of Security Functional Requirements:**
- Provide authentication
- Provide non-repudiation
- Audit: monitor and report
- Crypto support: key management
- Protect security functions
- User Data Protection: integrity protection, confidentiality
- Security Management: define roles, manage functions & data
- Privacy: protect against id discovery and misuse
- Ensure Availability: e.g. fault-tolerate architecture
- Provide trusted communication path between users & TSF
The Common Criteria

Process:

**Security Assurance Requirements:**
- Demonstrate that the PP is complete, sound, consistent
- Demonstrate that the ST is complete, sound, consistent and suitable for use as the basis for a ToE
- Ensure the delivery, installation, generation, initialization of the ToE
- Develop several levels of specification and design, evaluate consistency between levels
- Develop proper and complete documentation
- Ensure security integrity is not interrupted by maintenance or changes
- Analyze the existence of latent vulnerabilities (bypass authentication, covert channel, configuration weakness)
- Adequate test coverage, test depth, independent testing
- Configuration management: ToE developed, refined, modified
# The Common Criteria

## Process:

<table>
<thead>
<tr>
<th>TOE Security Specifications</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TSF - TOE Security Functions</strong></td>
<td>As set security functions for all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.</td>
</tr>
<tr>
<td><strong>SOF - Strength of Functions</strong></td>
<td>Qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behavior by directly attacking its underlying security mechanisms.</td>
</tr>
<tr>
<td><strong>TSC - TSF Scope of Control</strong></td>
<td>The set of interactions that can occur with or within a TOE and are subject to the rules of the TSP.</td>
</tr>
<tr>
<td><strong>TSFI - TOE Interface</strong></td>
<td>Set of interfaced, whether interactive (man-machine interface) or programmatic (application programming interface), through which TOE resources are accessed, mediated by the TSF, or information is obtained from the TSF.</td>
</tr>
</tbody>
</table>
The Common Criteria

Process:

<table>
<thead>
<tr>
<th>TOE Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>The realization of a TOE in accordance with its specifications.</td>
</tr>
</tbody>
</table>

Product meets a baseline set of security criteria and/or processes that institute fundamental security techniques.
The Common Criteria

Evaluation Levels:

EAL1: Functionally Tested
TOE functions in a manner consistent with its off-the-shelf documentation, and provides useful protection against identified threats

EAL2: Structurally Tested
Tested functionally using off-the-shelf documentation plus design information that is provided by the developer

EAL3: Methodically Tested and Checked
Functionally tested with more insight into the design and deeper test coverage. Developer provides evidence of a search for flaws

EAL4: Methodically Designed, Tested and Reviewed
Functionally tested with even more insight into the design and more test coverage. Testing supported by Independent labs searching for vulnerabilities
The Common Criteria

Evaluation Levels:

EAL5: Semiformally Designed and Tested
Product must have been developed using a rigorous methodology. Some formal methods, covert channel analysis and modular design are present. Independent search for possible exploits by attackers of moderate potential for success.

EAL6: Semiformally Verified Design and Tested
Formal methods and systematic covert channel analysis required. Product must be modular and layered in design.

EAL7: Formally Verified Design and Tested
More formal methods and systematic covert channel analysis required. Product must be modular and layered in design. Independent search for vulnerabilities by attacker with high attack potential is accomplished.
The Common Criteria

Evaluation Levels:

<table>
<thead>
<tr>
<th>Level</th>
<th>Requirements</th>
<th>Functional Specification</th>
<th>High-level Design</th>
<th>Low-level Design</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>EAL1</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
</tr>
<tr>
<td>EAL2</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
</tr>
<tr>
<td>EAL3</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
</tr>
<tr>
<td>EAL4</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
<td>Informal</td>
</tr>
<tr>
<td>EAL5</td>
<td>Formal</td>
<td>Semiformal</td>
<td>Semiformal</td>
<td>Informal</td>
<td>Informal</td>
</tr>
<tr>
<td>EAL6</td>
<td>Formal</td>
<td>Semiformal</td>
<td>Semiformal</td>
<td>Semiformal</td>
<td>Informal</td>
</tr>
<tr>
<td>EAL7</td>
<td>Formal</td>
<td>Formal</td>
<td>Formal</td>
<td>Semiformal</td>
<td>Informal</td>
</tr>
</tbody>
</table>
Protection Profiles/Security Policies & Targets

● **Example PPs:**
  https://www.cs.utexas.edu/~byoung/cs361/lecture78.pdf

● **Example Security Policies:**
  https://www.sans.org/security-resources/policies/

● **Microsoft Windows Security Target (CC)**
  http://gauss.ececs.uc.edu/Courses/c6055/pdf/microsoft-cc.pdf
References

https://www.cs.purdue.edu/homes/ninghui/readings/AccessControl/landwehr_etal_84.pdf

Bell-LaPadula model
https://wiki.umn.edu/CBI_ComputerSecurity/PubBellLaPadula

Biba model
https://wiki.umn.edu/CBI_ComputerSecurity/PubBibaModel

SANS Institute Protection Profiles
http://gauss.eceecs.uc.edu/Courses/c6055/pdf/SANS-CC-PP.pdf