Composite Evaluation for Smart Cards and Similar Devices

Dr. Igor Furgel
ISCI-WG1 and T-Systems GEI GmbH
What are we speaking about?

- Motivation
- Terminology and Scope
- Roles, Input and Information Transfer

- Composite Evaluation Approach
- Composite-specific Assurance Requirements

- Benefits of this Approach
Motivation

In the beginning there were
- Increasing complexity of IT products, and
- Need to master it.

How can I cope with a complex IT product?

Romans knew:

Divide et impera!
Motivation

Final IT products consist of different (hard- and software) components being produced by different manufacturers.

The component manufacturers wish to keep the most possible independency from each other.

They try to use well-defined interfaces of different kinds: technical, procedural, security.

A CC security certificate and ‘ETR for Composition’ are well-defined security interfaces.

But how can we use it?
Motivation

We developed a holistic concept for the composite product evaluation incl. the evaluation methodology (for CC v. 2.x and v. 3.1)
This approach is compliant with any EAL package
It is intended to represent this approach in form of a JIL- and of a CC Supporting Document

The aim of this contribution is to give
  – developers and
  – evaluators

a guidance

what relevant aspects have to be described and considered in the context of a composite evaluation

What is a composite evaluation?
Terminology & Scope

A **composite product** consists of at least two different parts, whereby one of them represents a single product having already been evaluated/certified.

The **composite TOE** comprises the whole composite product, i.e. the certified product is declared to be part of the composite TOE.

An evaluation of the composite TOE is a **composite evaluation**.
Terminology & Scope

- Usually, a composite product consists of two components, whereby the first one represents an **underlying platform** (‘Server’) and the second one constitutes an **application** (‘Client’) running on this platform. The underlying platform is usually the part of the composite product having already been evaluated.

<table>
<thead>
<tr>
<th></th>
<th>Smart card</th>
<th>Java</th>
<th>Crypto-box</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>application</strong></td>
<td></td>
<td>Operating system</td>
<td>Special crypto-box application (e.g. DigSign-Application)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Java applet</td>
<td></td>
</tr>
<tr>
<td><strong>underlying platform</strong></td>
<td></td>
<td>Integrated circuit</td>
<td>Hardware + boot-loader + core operating system</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Java run-time environment</td>
<td></td>
</tr>
</tbody>
</table>
Roles & Information Transfer
Composite Evaluation Approach

Two suitable types of CC-conform ‘requirement constructs’ for the current aim are:

– assurance package and
– refinements of the existing assurance requirements.

Both of them possess an appropriate abstraction level being independent of concrete products and product families.

We decided in favour of the refinements in order to retain conformance to the EAL packages.

These refinements describe how the existing work units should be applied to the composite issue.
Composite Evaluation Approach

- We defined
  - composite evaluation concept,
  - composite evaluation activities,
  - ETR for composite evaluation,
  - composite-specific assurance requirements meeting the current objective and
  - the evaluation methodology (evaluator actions) for these assurance requirements.

- This methodology mostly is independent of a CC version and, thus, applicable for CC v2.x as well as for CC v3.x.
Composite Evaluation Approach

The CC Supporting Document for Composite Evaluation defines the following composite-specific refinements of assurance requirements (as example for CCv.3.1):

<table>
<thead>
<tr>
<th>Composite Evaluation Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASE_COMP</td>
<td>Consistency of Composite Product Security Target</td>
</tr>
<tr>
<td>ALC_COMP</td>
<td>Integration of Composition Parts and Consistency of delivery procedures</td>
</tr>
<tr>
<td>(v.2.x: ACM_COMP + ADO_COMP)</td>
<td></td>
</tr>
<tr>
<td>ADV_COMP</td>
<td>Composite Design Compliance</td>
</tr>
<tr>
<td>ATE_COMP</td>
<td>Composite Functional Testing</td>
</tr>
<tr>
<td>AVA_COMP</td>
<td>Composite Vulnerability Assessment</td>
</tr>
</tbody>
</table>
ASE_COMP: Consistency of the Composite Security Target

The aim of this activity is to determine whether

the Security Target of the \textit{composite product} does not contradict
the Security Target of the \textit{underlying platform}.

‘Three steps technology’ for the Composite-ST:

– Step 1: \textit{Formulate a Security Target} of the composite product using the standard code of practice. The Composite-ST can be formulated independent of the Platform-ST.

– Step 2: \textit{Determines the intersection} of the Composite-ST and the Platform-ST by analysing and comparing their SFR/TSF.

– Step 3: \textit{Determines under which conditions he can trust in and rely on the Platform-TSF} being used by the Composite-SP without a new examination.
ASE_COMP: General Scheme

Walk up-right-down through the structure of the Security Target of the platform

- Threats
- Org. Sec. Policies
- Assumptions

rationale ⇒ TOE Objectives ⇒ Objectives for Environment

rationale ⇒ TOE Security Functional Requirements ⇒ TOE Security Assurance Requirements

rationale ⇒ TOE Security Functionality (TSF) ⇒ Assurance Measures (AM)

Is not defined in v.3.x
ASE_COMP: Relevant Platform SFR

Before you go up: Determine the intersection relevant PSF (Platform Security Functions) that have to be considered further:

- If the Composite-ST does not use any property of the Platform-ST and, hence, the intersection Relevant Platform SFR is an empty set, no further composite evaluation activities are necessary. **In such a case there is a technical, but not a security composition.**

Example: smart card operating system on an integrated circuit card
- used HW features:
  - RNG, AES and RSA,
  - but not DES
ASE_COMP: Significant Platform Assumptions

Before you go down: Determine the **significant PA** (Platform Assumptions) having to be considered further:

- **Composite-fulfilled PA**: The composite does it
- **irrelevant PA**
- **Significant PA**: Composite’s environment has to care
How can we decide that the degree of trustworthiness of the Relevant Platform SFR/TSF is sufficient for the composite evaluation?

We shall compare the Platform-AR (Assurance Requirements) with the Composite-AR.

The degree of trustworthiness of the Platform-TSF is sufficient, if

\[
\text{Platform AR} \supseteq \text{Composite AR}
\]

It is fulfilled, for example, if

\[
\text{Platform EAL} \supseteq \text{Composite EAL}
\]
The aims of this activity are to determine

–whether the correct version of the application is installed onto/into the correct version of the underlying platform and

–whether the delivery procedures of Platform and Application Developers are compatible with the acceptance procedure of the Composite Product Integrator.
The aim of this activity is to determine

– whether the requirements on the application, imposed by the underlying platform, are fulfilled in the composite product.

Such requirements on the application can be formulated in the relevant certification report, user guidance and ETR_COMP for the platform.

The thread for this activity is given in the Design Compliance Evidence provided by the Application Developer.
ATE_COMP

- The aim of this activity is to determine
  - whether composite product as a whole exhibits the properties necessary to satisfy the functional requirements of its Security Target.

- Integration testing means that the composite product is being tested as it is: the application is running on the platform.

- Behaviour of implementation of some SFRs can depend on properties of the underlying platform as well as of the application (e.g. correctness of the implementation of tamper resistance against physical attacks).

- This evaluator activity is the complementary part to ASE_COMP, where he determined, on which part of the Platform-ST the Composite-ST can rely (the group Relevant Platform SFR);
  - In ATE_COMP he determines, whether the Composite-ST can also rely on the platform’s functional behaviour being not covered by the Platform-ST.
The aim of this activity is to determine
- the exploitability of flaws or weaknesses in the composite TOE as a whole in the intended environment.

Vulnerability analysis shall be confirmed by penetration testing of the final composite TOE.

The platform may also be used in a way that was not foreseen by the Platform Developer and Platform Evaluator, or the Application Developer may not have followed the stipulations provided with the platform certification. In such cases:
- The Composite Product Evaluator can consult the Platform Evaluator and draw on his experience gained during the platform evaluation.
- Separation of vulnerabilities of application and platform with the use of “open samples” in order to use test software without the application countermeasures, whereby any platform inherent countermeasures are still active.
AVA_COMP: ETR_COMP

- The results of the vulnerability assessment for the underlying platform represented in the ETR_COMP can be reused, if they are up to date and all composite activities for correctness – ASE_COMP, ALC_COMP, ADV_COMP and ATE_COMP – are finalised with the verdict PASS.

- The content and the structure of ETR_COMP has been reconciled with JIHAS.

- The ETR_COMP document is compiled from the ETR by the Platform Evaluator and must be approved by all parties involved in the platform evaluation.

- The content of ETR_COMP has to strike the right balance between protecting Platform Developer’s and/or Platform Evaluator’s proprietary information and providing sufficient information for the Composite Product Evaluator and the respective Certification Body.
AVA_COMP: Content of the ETR_COMP

- The information required within an ETR_COMP is focused on:
  - Formal information about the platform like its exact identification, reference to the certification report etc,
  - Information about the Platform design,
  - Information about the evaluated configuration of the Platform,
  - Information on delivery procedures and data exchange,
  - Information about penetration testing of the Platform including the considered attack scenarios and summary of test results.
  - Observations and recommendations for users (beyond the platform user guidance):
    - Observations on the evaluation results (e.g. specific TOE configuration for the evaluation),
    - Recommendations/stipulations for the Composite Product Evaluator: specific information on use of the evaluation results (e.g. about specific testing necessary during a composition evaluation).
Evaluation Results and Final Security Certificate

- Results of a composite evaluation will be provided to the Composite Product Certification Body in form of an Evaluation Technical Report for the composite product with the overall verdict for the composite evaluation based on the partial verdicts for each assurance component in scope, e.g. within an EAL package.

- As the composite product certificate covers also the platform, the composite product certificate validity is linked to the validity of the platform certificate. The question of certificate’s validity for the current composite product certification also includes the determination of equivalence of single assurance components (and, hence, of assurance levels) belonging to different CC versions, if applicable.

- Composite Product Certification Body issues a security certificate for the composite product considering the final verdict and validity of the platform certificate.
Current Composite Approach vs. CAP-Package:
Definition of Composite vs. Composed TOE

Current Approach with EAL-Refinement

- Only platform is certified
- Evaluation of composite TOE = evaluation of application + reuse of platform evaluation results

Package CAP

- Dependent TOE and Base TOE are certified
- Evaluation of composed TOE = evaluation of interaction + reuse of base evaluation results + reuse of dependent evaluation results
Current Composite Approach vs. CAP-Package: Assurance

Current Approach with EAL-refinement

- Assurance level is **described by EAL with augmentation**
- Assurance requirements: composite TOE \( \subseteq \) platform TOE
- supporting document with methodology description on the level of detail ‘work unit’
- platform evaluation produces additional ‘ETR for Composition’; no internal details of the platform are necessary
- Concept of composite evaluation does not limit the composite evaluation in EAL and resistance against attacks, i.e. up to ‘high’

Package CAP

- Assurance of CAP is **not comparable with EAL**
- Assurance for interfaces: dependent TOE \( \leq \) base TOE
- class ACO and assurance packages CAP-A, CAP-B and CAP-C
- guidance for the application of the families ACO_DEV and ACO_REL
- no specific evaluation results of the component evaluation for the evaluation of the composed product
- Assurance packages CAP is limited by resistance against attacks ‘**extended-basic**’ (ACO_VUL.3)
Benefits of the Composite Approach

- Universally applicable
  - to all EAL packages and all attack potentials
  - to all kinds of composite products and various CC versions
- Clear alignment with the actual security features of the underlying platform by justification of the composite product’s Security Target
- Minimised risk of getting incompatibility problems in a very late evaluation phase (e.g. vulnerability analysis or ETR), since compatibility is checked as early as possible
- Standardised approach by definition of the composite EAL-Refinement and the methodology proposed
- Using ‘ETR for Composition‘ as an universal well-defined ‘security interface‘
- Cost reduction by excluding evaluated parts of a composite product