

Security Target in a Crossfire of
Precision and Readability:
What a Vulnerability Analysis shall
refer to?

Igor Furgel
T-Systems



Motivation: What is the issue?

- A Security Target (ST) as a definition of a TOE Security Policy aims two contradictory purposes:
 - to be as precise and unambiguous as possible, and
 - to be readable and understandable for a potential consumer of the TOE.
- Simultaneously, there is still a discussion within the CC community, what a **vulnerability analysis** should **address** within the ST:
 - exclusively **security functional requirements** or also **security objectives**?
- The aim of this contribution is
 - to outline different approaches for the construction of a ST impacting these issues,
 - to analyse their **pro** and **contra**,
 - to give some hints for the current and future content of a ST,
 - to infer an answer what a vulnerability analysis should refer to in a ST.



Security Target: Major Units

Security Problem Definition (**SPD**):
What is the security problem?

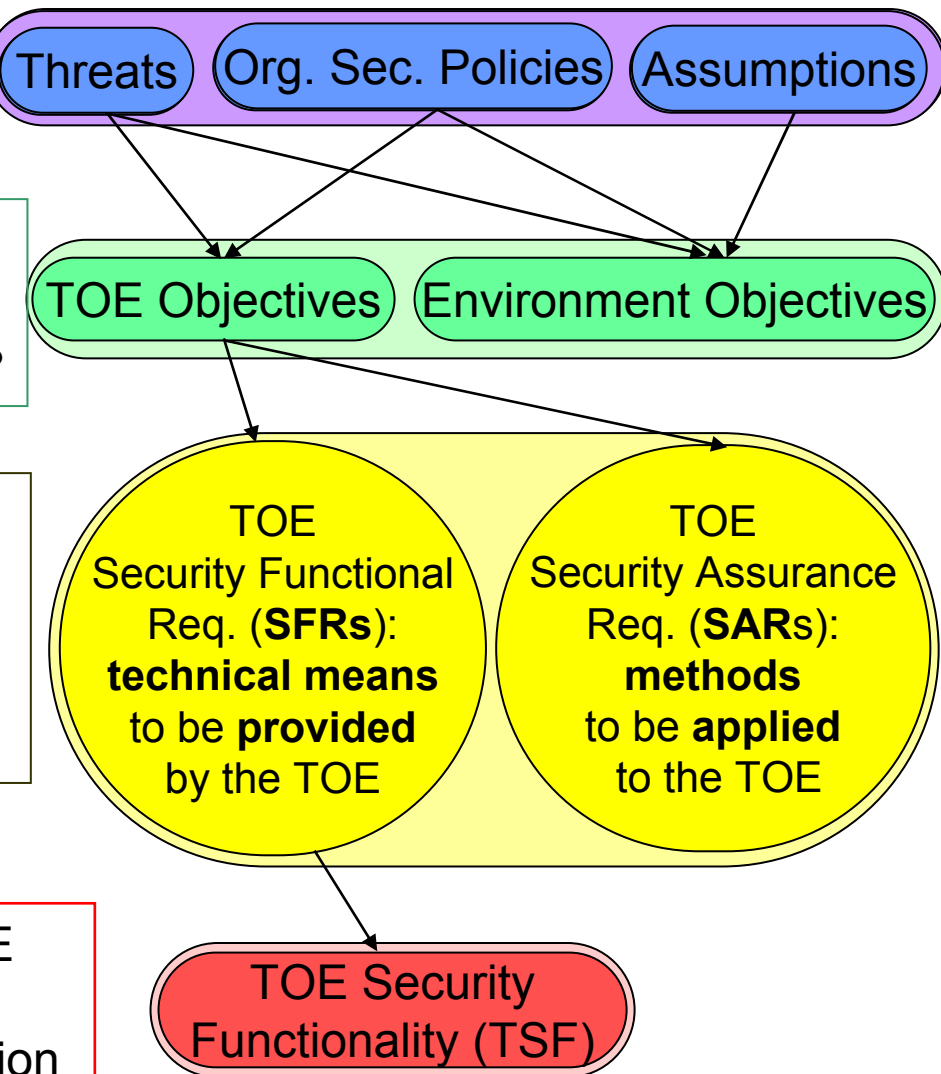
Security Solution Definition (**SSD**):
How do we intend to solve / to cope with the current security problem?

Security Objectives (**SO**):
What is to achieve?

Security Requirements (**SR**):
By which means are the SOs to achieve?

Solution Implementation Definition (**SID**): **By which measures do we intend to implement this security solution?**

TSS: TOE summary specification



'Precision' and 'Readability' – How can the balance be justified?

- An optimal balance between the poles '*precision*' and '*readability*' can especially be justified within Security Solution Definition (i.e. Security Objectives + Security Requirements). There can be the following approaches:
 - ITSEC-like: only SOs without SFRs,
 - CC-standard: SFRs are partially defined,
 - CC-enhanced: SFRs are fully defined, and
 - SFRs-only: Full definition of SFRs without SOs.
- We will now consider these approaches with related advantages and disadvantages.



(1) ITSEC approach

- A ST merely contains Security Objectives (TOE + environment) as an informal description, but no SFRs.
- Advantage: The ST is
 - **well readable and understandable,**
 - easy to create.
- Disadvantage:
 - The ST is more ambiguous, less precise => several interpretations might undermine the Security Policy aimed by the ST;
 - **The only criterion for the vulnerability analysis is here achieving SOs. It causes a quite big ambiguity inherited from the informal character of the SO-statement.**



(2) CC-standard approach (1/3)

- A ST states informal SOs (TOE + environment) and semiformal SFRs, whereby the SFRs have the following semantic structure:
 - SFR == {functionality required} || {dependencies} || {hierarchy}.
- This semantic structure does not unconditionally include any explicit security purpose of an SFR / security contribution to the part of Security Policy required by an SFR.
- Example: FCS_COP.1 ‘Cryptographic operation’:
 - The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].
- This SFR does not state, what shall concretely be secured (asset and its security property). **Hence, it cannot serve as a criterion for vulnerability analysis.**



(2) CC-standard approach (2/3)

- Advantage:
 - The ST is still sufficiently **well readable and understandable** due to an informal SO-statement.
 - **Significant reducing interpretation opportunities** owing to a semiformal SFR-statement.
 - **Big flexibility in mapping SFRs to SOs**, so that one SFR can contribute to achievement of several SOs.
- Disadvantage:
 - The semantic structure of the SFRs **does not enforce** any statement of an SFRs-attribute showing **what the contribution of this SFR to the achievement of the SOs is**.
It means that it randomly depends on a concrete formulation of this or those SFR (cf. e.g. FCS_COP.1 and FPT_ITC.1).



(2) CC-standard approach (3/3)

- The major question of an evaluator here is: what should I use as the criterion for the vulnerability analysis (VAN) – **SFRs** or **SOs** or **both**?
- CC v3.1, rev. 3 unequivocally say: ‘SFRs’!
- But this decision, in fact, depends very much and randomly on a concrete formulation of this or those SFR:
 - If an SFR states merely the required functionality without any belonging security purpose (as done e.g. in the class FCS), the evaluator shall also consider the related SOs as the criterion for VAN, because it is mostly impossible to decide on enforcing a Security Policy just looking at pure functionality.



(3) CC-enhanced approach (1/2)

- A ST comprises informal SOs (TOE + environment) and semiformal SFRs, whereby the SFRs have a **full** semantic structure:
 - SFR == {functionality required} || **{security purpose to be covered}** || {dependencies} || {hierarchy}
- The attribute **{security purpose to be covered}** exactly shows the security contribution of this SFR to the current Security Policy.
- Example: **FCS_COP.1/full_semantic** ‘Cryptographic operation’:
 - The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*] **in order to protect [selection: *confidentiality, integrity, authenticity, [assignment: list of other security properties]*] of [assignment: list of assets]**.



(3) CC-enhanced approach (2/2)

- Advantage:
 - The ST is still sufficiently **well readable and understandable** due to an informal SO-part.
 - **Maximal reducing interpretation opportunities** owing to the FULL semiformal SFR-statement.
 - Due to the full definition, each SFR is self-sufficient: It also comprises – per semantic rules – the SFR-attribute **{security purpose to be covered}** showing what the contribution of this SFR to the achievement of the SOs is.
 - The mapping between the SOs and SFRs can be followed in a very precise way.
 - **The evaluator can use ONLY SFRs (without looking at the SOs) as the criterion for the VAN.**
- Disadvantage:
 - **Less flexibility in mapping the SFRs to the SOs:** it might be necessary to instantiate (to iterate) a dedicated SFR for each SO, to whose achievement this SFR contributes.



(4) SFRs-only approach (1/2)

- A ST states no SOs (TOE + envir.) at all, but only security requirements on environment and semiformal SFRs, whereby the SFRs have the already known full semantic structure:
 - SFR == {functionality required} || **{security purpose to be covered}** || {dependencies} || {hierarchy}
- The attribute **{security purpose to be covered}** - like in the CC-enhanced approach - exactly shows the security contribution of this SFR to the current Security Policy.
- In order also to address the environment, the security requirements for the environment shall be defined, too.



(4) SFRs-only approach (2/2)

- Advantage (similar to the CC-enhanced approach):
 - Maximal reducing interpretation opportunities owing to the full semiformal SFR-statement.
 - Due to the full definition, each SFR is self-sufficient: It also comprises – per semantic rules – the SFRs-attribute {security purpose to be covered} showing what the contribution of this SFR to the achievement of the SOs is.
 - **The evaluator can use ONLY SFRs (without looking at the SOs) as the criterion for the VAN.**
 - Less effort in creating an ST due to absence of SOs and the rationale SOs <-> SFRs.
- Disadvantage:
 - **Badly readable and understandable** due to the absence of the informal SO-part.



Results (1/2)

- The current contribution shows that the approaches
 - (1) ITSEC-like and
 - (4) SFRs-only
- **offend** the purposes of a ST (precision AND readability).

- Hence, the **only** approaches
 - (2) CC-standard and
 - (3) CC-enhanced
- may be considered as **sound** in the sense of optimising the balance between being
 - as precise and unambiguous as possible and
 - readable and understandable for a potential consumer of the TOE.



Results (2/2)

- The **approach (2) ‘CC-standard’** offers
 - a **bigger flexibility**,
 - but **less unambiguity** and clearness concerning the decision criterion for vulnerability analysis (VAN).
Hence, **SOs shall also be here addressed / used as the VAN criterion. It may cause more evaluation effort in performing vulnerability analysis.**
- The **approach (3) ‘CC-enhanced’** offers
 - the **maximal sensible unambiguity** and clearness concerning the decision criterion for VAN, **whereby SFRs by themselves are sufficient as the VAN criterion**,
 - but **less flexibility** and **more effort in writing a ST.**
- The CC community may **endorse both approaches** – the current one and of fully-defined SFRs – and let the final decision to the author of a ST or PP.



Thank you for your attention!

Dr. Igor Furgel

T-Systems International

SSC Security

**Vorgebirgsstr. 49
53119 Bonn**

 **+49 228 9841-5120**

 **igor.furgel@t-systems.com**

