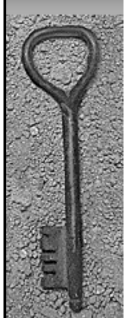
Operational Assurance

- Role of Operational Assurance
- Operational Assurance in the CC
  - –Class ADO – Delivery and Operation
  - –Class AGD – Guidance Documentation
  - –Class AVA – Vulnerability Assessment

600.436 High-Assurance Systems Fall 01/Operational Assurance 1



Role of Operational Assurance

- As evaluation criteria, CC concentrates on activities during TOE development and evaluation
- Current CC assurance classes don’t really address Operations, but they do consider hand-off from developer to user/operator
- There are several efforts to produce assurance classes that assign evaluatable requirements to operator, administrator, and user activities

600.436 High-Assurance Systems Fall 01/Operational Assurance 2

Operational Assurance in the CC

- CC institutionalizes Operational Assurance through 3 Assurance Classes containing 8 Assurance Families
  - Delivery and Operation (ADO)
    - Delivery; Installation, Generation, and Startup
  - Guidance Documents (AGD)
    - Administrator Guidance; User Guidance
  - Vulnerability Assessment (AVA)
    - Covert Channel Analysis; Misuse; Strength Of TOE Security Functions; Vulnerability Analysis

Delivery and Operation (ADO)

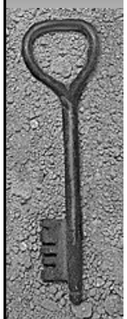
- Goal of Delivery and Operation is ensure that the TOE that customers operate is within the range of configurations that were evaluated.
- The two Families in the Class are:
  - Delivery (ADO_DEL)
  - Installation, Generation and Startup (ADO_IGS)

Delivery

- Focus is on ensuring that the TOE received by a customer is exactly the TOE that was shipped from the developer
- Three Components, levelled on the basis of increasing requirements on the developer to detect and prevent modifications to the TOE during delivery.
  - Delivery Procedures
  - Detection of Modification
  - Prevention of Modification

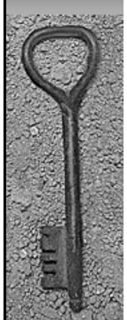
Delivery Procedures

- **The developer shall document procedures for delivery of the TOE or parts of it to the user**
  - **The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site**
- **The developer shall use the delivery procedures**



Detection of Modification

- The developer shall document procedures for delivery of the TOE or parts of it to the user
  - **The delivery documentation shall describe how the various procedures and technical measures provide for the detection of modifications, or any discrepancy between the developer’s master copy and the version received at the user site**
  - **The delivery documentation shall describe how the various procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user’s site**



Prevention of Modification

- The developer shall document procedures for delivery of the TOE or parts of it to the user
  - The delivery documentation shall describe how the various procedures and technical measures provide for the **prevention of modifications**, or any discrepancy between the developer's master copy and the version received at the user site

Installation, Generation and Startup

- Goals of Installation, Generation and Startup are to ensure that the TOE has been installed, generated, and started up in a secure manner as intended by the developer
  - i.e., a secure transition from the TOE's implementation representation being under configuration control to its initial operation in the user environment
- Two Components, levelled on the basis of whether the TOE generation options are logged
  - Installation Generation and Start-up (IGS) Procedures
  - Generation Log

IGS Procedures

- **The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE**
  - **describe the steps necessary for secure installation, generation, and start-up of the TOE**

Generation Log

- The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE
  - **describe procedures capable of creating a log containing the generation options used to generate the TOE in such a way that it is possible to determine exactly how and when the TOE was generated**

Guidance Documents (AGD)

- Goal of Guidance Documents is to ensure that the developer provides effective guidance to both administrators and users on their responsibilities for TOE security
- The two Families in the Class are:
  - Administrator Guidance
  - User Guidance

Administrator Guidance

- Goal of Administrator Guidance is to help administrators understand the security functions provided by the TOE, including both those functions that require the administrator to perform security-critical actions and those functions that provide security-critical information
- One Component
  - Administrator Guidance

600.436 High-Assurance Systems Fall 01/Operational Assurance 13

Administrator Guidance

- **The developer shall provide administrator guidance addressed to system administrative personnel that shall:**
  - **describe the administrative functions and interfaces available to the administrator of the TOE**
  - **describe how to administer the TOE in a secure manner**
  - **contain warnings about functions and privileges that should be controlled in a secure processing environment**
  - **describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE**
  - **describe all security parameters under the control of the administrator, indicating secure values as appropriate**
  - **describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF**
  - **be consistent with all other documentation supplied for evaluation**
  - **describe all security requirements for the IT environment that are relevant to the administrator**

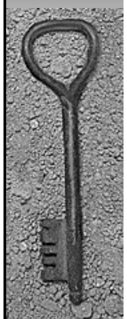
600.436 High-Assurance Systems Fall 01/Operational Assurance 14

User Guidance

- Goal of User Guidance is to describe the security functions provided by the TSF along with instructions and guidelines, including warnings, for their secure use by non-administrative users
- One Component
  - User Guidance

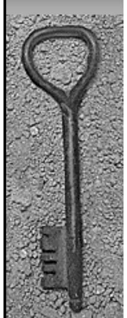
User Guidance

- **The developer shall provide user guidance that shall:**
  - **describe the functions and interfaces available to the non-administrative users of the TOE**
  - **describe the use of user-accessible security functions provided by the TOE**
  - **contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment**
  - **clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment**
  - **be consistent with all other documentation supplied for evaluation**
  - **describe all security requirements for the IT environment that are relevant to the user**



Vulnerability Assessment (AVA)

- Goal of Vulnerability Assessment is to assess residual TOE security risks arising from
  - intrinsic limitations of chosen security mechanisms
  - misuse or misconfiguration of chosen security mechanisms
  - flawed implementation of chosen security mechanisms
- The four Families in the Class are:
  - Covert Channel Analysis (AVA_CCA)
  - Misuse (AVA_MSU)
  - Strength of TOE Security Functions (AVA_SOF)
  - Vulnerability Analysis (AVA_VLA)



Covert Channel Analysis

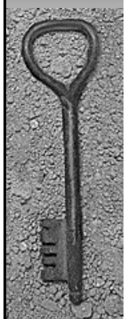
- Goal of Covert Channel Analysis is to determine the existence and potential capacity of unintended signalling channels (i.e. illicit information flows) that may be exploited
  - Only applies if there are information flow SFRs
- Three Components, levelled based on the rigor of the covert channel analysis
  - Covert Channel Analysis
  - Systematic Covert Channel Analysis
  - Exhaustive Covert Channel Analysis

Covert Channel Analysis

- **The developer shall conduct a search for covert channels for each information flow control policy.**
- **The developer shall provide covert channel analysis documentation that shall:**
  - **identify covert channels and estimate their capacity**
  - **describe the procedures used for determining the existence of covert channels, and the information needed to carry out the covert channel analysis**
  - **describe all assumptions made during the covert channel analysis**
  - **describe the method used for estimating channel capacity, based on worst case scenarios**
  - **describe the worst case exploitation scenario for each identified covert channel**

Systematic Covert Channel Analysis

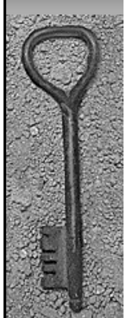
- The developer shall conduct a search for covert channels for each information flow control policy.
- The developer shall provide covert channel analysis documentation that shall:
  - **<u>provide evidence that the method used to identify covert channels is systematic</u>**
    - Primary additional requirement here is that there be a systematic strategy for identifying and verifying covert channels



Exhaustive Covert Channel Analysis

- The developer shall conduct a search for covert channels for each information flow control policy.
- The developer shall provide covert channel analysis documentation that shall:
  - provide evidence that the method used to identify covert channels is **exhaustive**
    - “exhaustive” is meant to imply that there are strong arguments that following the CCA plan will uncover all covert channels that might exist

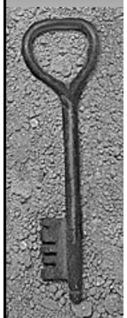
600.436 High-Assurance Systems Fall 01/Operational Assurance 21



Misuse

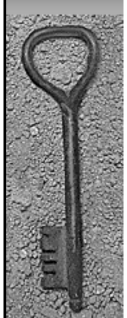
- Goal of Misuse is to determine whether the TOE can be configured or used in a manner that is insecure but that an administrator or user of the TOE would reasonably believe to be secure
- Three Components, levelled on the basis of increasing evidence provided by the developer and the increasing rigour of analysis
  - Examination of Guidance
  - Validation of Analysis
  - Analysis & Testing for Insecure States

600.436 High-Assurance Systems Fall 01/Operational Assurance 22



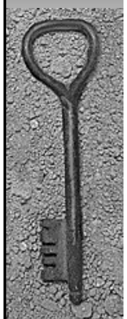
Examination of Guidance

- The developer shall provide guidance documentation that shall:
  - identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation
  - be complete, clear, consistent and reasonable
  - list all assumptions about the intended environment
  - list all requirements for external security measures (including external procedural, physical and personnel controls)



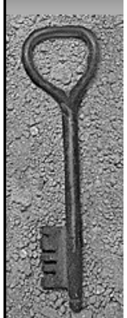
Validation of Analysis

- The developer shall provide guidance documentation
- **The developer shall document an analysis of the guidance documentation**
  - **The analysis documentation shall demonstrate that the guidance documentation is complete**



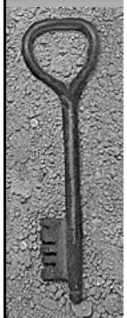
Analysis & Testing for Insecure States

- No additional explicit developer requirements
- Evaluator is required to perform independent testing to verify that the guidance documentation provides enough information to identify insecure TOE configurations



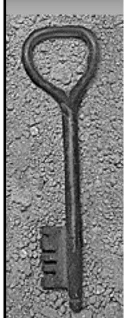
Strength of TOE Security Functions

- Goal of Strength of TOE Security Functions is to qualify the strength of security mechanisms that can be attacked using quantitative or statistical analysis techniques in terms of the effort required to successfully mount the attack
- One Component
  - Strength of TOE security function evaluation



Strength of TOE security function evaluation

- **For each mechanism identified in the ST as having a strength of TOE security function claim, the developer shall perform a strength of TOE security function analysis that shall show that:**
  - **Each mechanism meets or exceeds the minimum strength level defined in the PP/ST**
  - **Each mechanism meets or exceeds the specific strength of function metric defined in the PP/ST**



Vulnerability Analysis

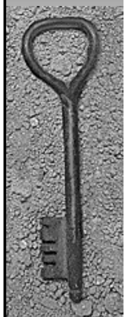
- Goal of Vulnerability Analysis is to determine if vulnerabilities identified - during the evaluation of the construction and anticipated operation of the TOE or by other methods (e.g. by flaw hypotheses) - could allow users to violate the TSP
- Four Components, levelled based on an increasing rigour of vulnerability analysis by the developer and the evaluator
  - Developer Vulnerability Analysis
  - Independent Vulnerability Analysis
  - Moderately Resistant
  - Highly Resistant

Developer Vulnerability Analysis

- **The developer shall perform and document an analysis of the TOE deliverables searching for obvious ways in which a user can violate the TSP**
  - **The documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE**
- **The developer shall document the disposition of obvious vulnerabilities**

Independent Vulnerability Analysis

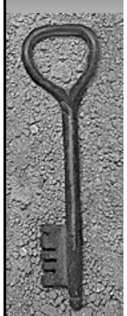
- The developer shall perform and document an analysis of the TOE deliverables searching for **ways** in which a user can violate the TSP
  - **The documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks**



Moderately Resistant

- The developer shall perform and document an analysis of the TOE deliverables searching for ways in which a user can violate the TSP
  - **The evidence shall show that the search for vulnerabilities is systematic**

600.436 High-Assurance Systems Fall 01/Operational Assurance 31



Highly Resistant

- The developer shall perform and document an analysis of the TOE deliverables searching for ways in which a user can violate the TSP
  - **The analysis documentation shall provide a justification that the analysis completely addresses the TOE deliverables**

600.436 High-Assurance Systems Fall 01/Operational Assurance 32