Analysis of the composition problems in CC v3.1 rev.1 with some suggested solutions

Dr. Albert B. Jeng and Yu-Min Yu
Telecom Technology Center, Taiwan

Sponsor ：National Communication Commission

20. September 2006 Spain

Outline

- Introduction
- The Generic Composition Problem
- Types of Composition
- How CC v3.1 rev.1 deals with the Composition Problem?
- How TNI/TDI Addresses the Composition Problem?
- Suggested Solution to Some of the Composition Problem
- Conclusion

Introduction (1/2)

- It is often that an IT solution is implemented by a variety of vendors by combining either evaluated or non-evaluated components
- The old CCv2.x does not provide a means of reusing the evaluation evidence and results of evaluations performed for the component developer
  - Instead, a new evaluation must be performed upon the combination

Introduction (2/2)

- In CC v3.1 rev.1, a new class on Composition was developed to address the issue that arises when a TOE includes a product that has been evaluated before (e.g., a database running atop an evaluated operating system)
- The new Composition class defines what needs to be done to leverage off the results of the existing CC evaluation

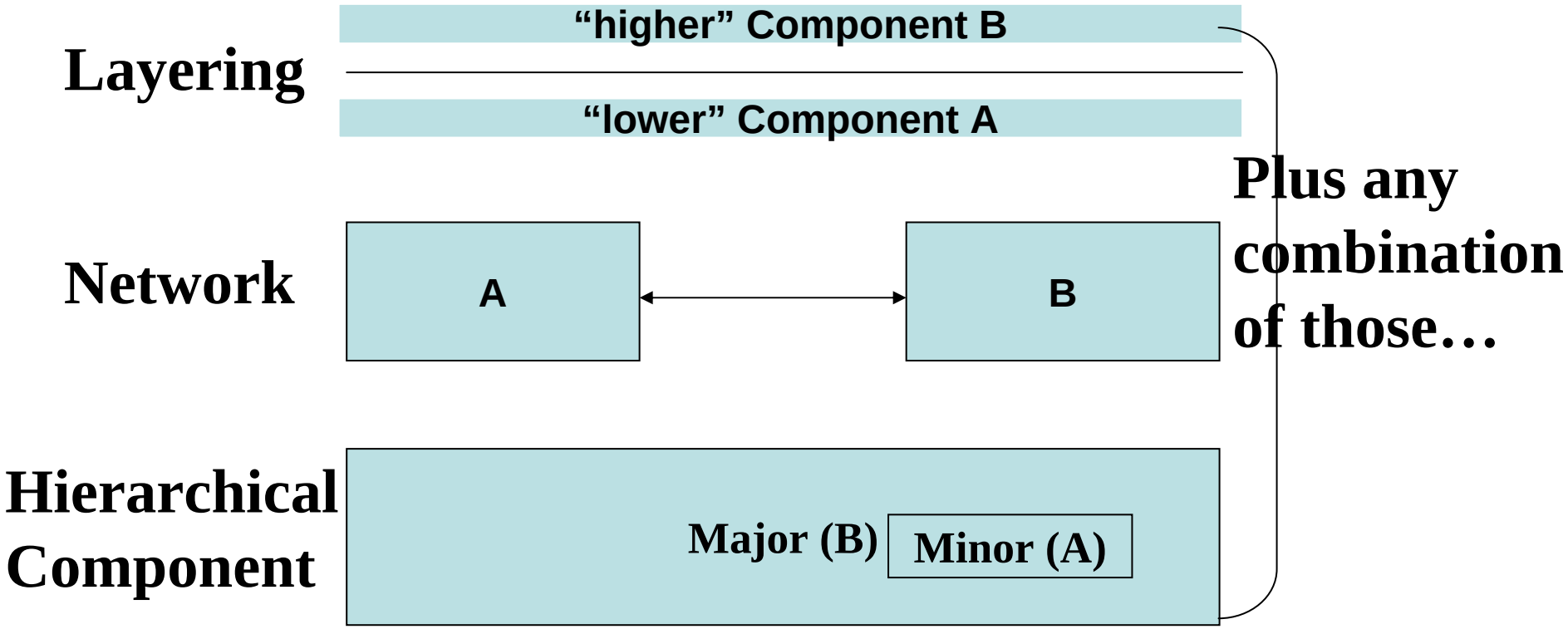
The Generic Composition Problem

- What happens when putting together the results of two individual component TOEs for operational use, with no further development?
- How to determine assurance of the composed TOE?
- How can a composed product be evaluated?
- How much can be re-used from the evaluation of individual components and what needs to be considered when re-using evaluation results?

Four Key Factors in a General Model for Composite Evaluations (Kurth & Karger ICCC5)

- The following factors comprise a general model for composite evaluations
  - the type of composition
  - the security policy the component evaluation was based on
  - the assurance level
  - the overall policy of the composite product

Types of Composition



(Adapted from Kurth and Karger ICCC5)

Composition Problem Areas

Composition Type	Composition Problem Areas
Layering	**the higher layer may depend on functions not considered to be secure functions in the evaluation of the lower layer**
Network	**when evaluating component A one states security requirements for the "other" component, but still** • **Security functions may not fit together** • **Assumptions made on a component may not be valid** • **Security functions may have unwanted side effects**
Hierarchical Component	**Due to the lack of separation, components may** • **Bypass security functions of the other components** • **Modify the security functionality and policy of other components and the whole product** • **Introduce a number of critical side effects**

(Extracted from Kurth and Karger ICCC5)

The CC Composition Evaluation

- Assume Component A has been evaluated
- Assume Component B shall be evaluated making use of the evaluation of A
  - What can be re-used and how?
  - What needs to be re-done?
- The CC Composition Class ACO provides a solution to the practical issues of adding evaluation results

Reusability of the Component Evaluation Result

- Network composition is the easiest
- Layered composition is more complex
- Hierarchical component composition is the most complex scenario
- In general, when the two individual component TSPs are “independent” and with “minimal interaction”, the composition is easier

Class ACO: Composition

- **Assurance class ACO: Composition defines requirements of the information necessary to ensure that two or more components, which have themselves been the subject of evaluation, can be integrated in a secure manner.**
- **The ACO: Composition assurance requirements will be applied to the composed TOE to:**
  - **a) determine that the required assurance is provided by the base component;**
  - **b) determine that the base component and dependent component are compatible; and**
  - **c) search for any vulnerabilities introduced through composing the base and dependent components into a single composed TOE entity.**

Class ACO: Composition encompasses five families (1/2)

- **Composition rationale (ACO_COR)**
  - The Composition rationale (ACO_COR) family is used to determine whether or not the appropriate assurance measures have been applied to the base for successful integration in the composed TOE.
  - That is, the SARs claimed by the base component are consistent with the SARs in the assurance package for the composed TOE.
- **Development evidence (ACO_DEV)**
  - Development evidence (ACO_DEV) provides details of the base component interfaces and internals in increasing levels of detail, mirroring the level of detail provided by Reliance of dependent component (ACO_REL).
- **Reliance of dependent component (ACO_REL)**
  - The Reliance of dependent component (ACO_REL) family considers the interactions between the components where the dependent component relies upon a service from the base component to support the operation of security functionality of the dependent component.

Class ACO: Composition encompasses five families (2/2)

- **Composed TOE testing (ACO_CTT)**
  - **This family requires that testing of composed TOE and testing of the base component, as used in the composed TOE, is performed.**
    - ✓ **a) testing of the interfaces between the base component and the dependent component, which the dependent component rely upon for enforcement of security functionality, to demonstrate their compatibility;**
    - ✓ **b) testing of the composed TOE to demonstrate that the TOE behaves in accordance with the SFRs for the composed TOE.**
- **Composition vulnerability analysis (ACO_VUL)**
  - **ACO_VUL includes determination of two different aspects of resistance by the composed TOE, namely:**
    - ✓ **a) Residual vulnerabilities in the base and dependent components remain unexploitable in the operational environment of the composed TOE;**
    - ✓ **b) The composed TOE is resistant to attackers with a given level of attack potential.**

Evaluation assurance level (EAL) and Composition Assurance Package (CAP)

EAL

- EAL1 - functionally tested
- EAL2 - structurally tested
- EAL3 - methodically tested and checked
- EAL4 - methodically designed, tested, and reviewed
- EAL5 - semiformally designed and tested
- EAL6 - semiformally verified design and tested
- EAL7 - formally verified design and tested

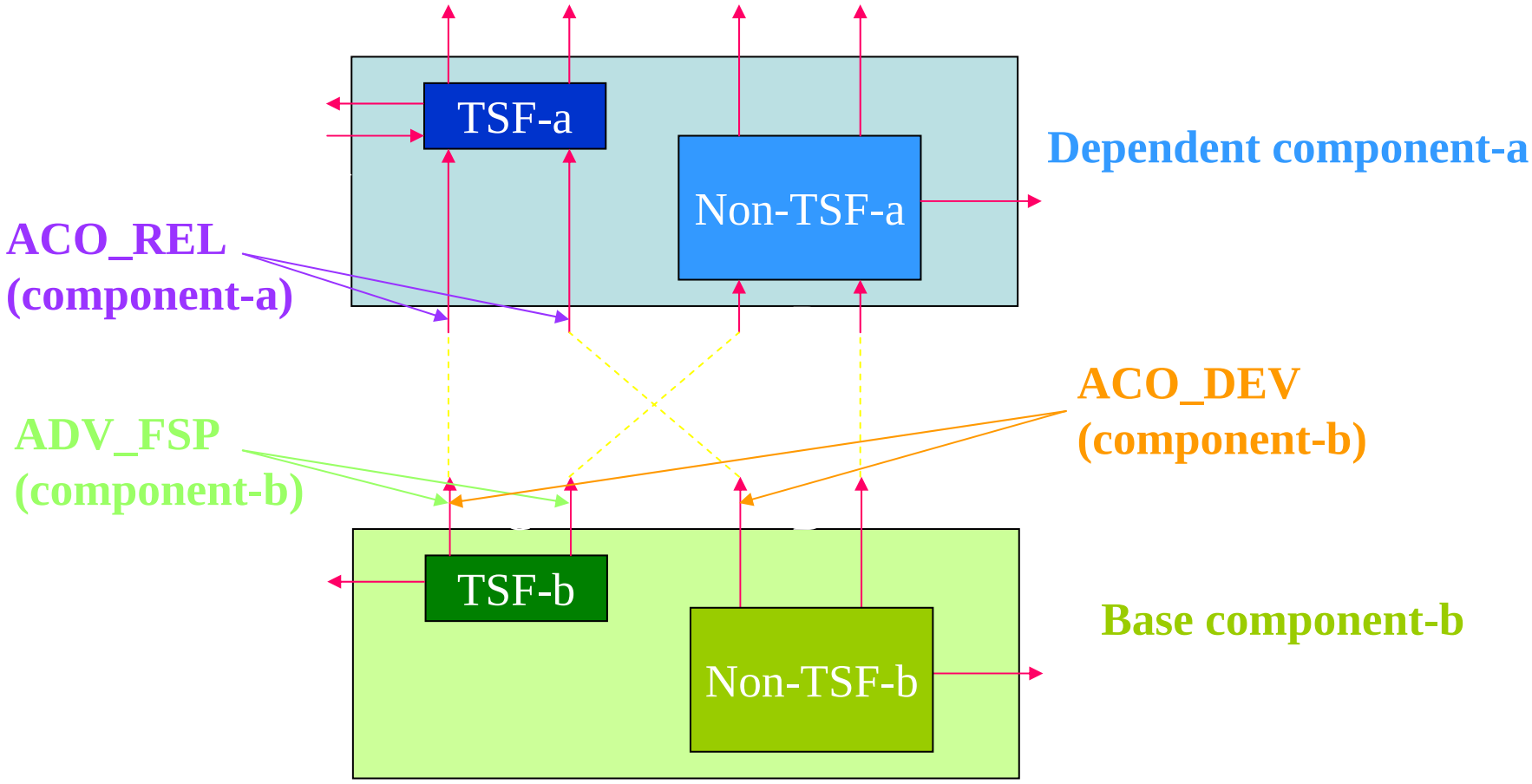
CAP

- CAP-A - Structurally composed
- CAP-B - Methodically composed
- CAP-C – Methodically composed, tested and reviewed

Composition Assurance Package (CAP) (CC v3.1 rev.1)

Assurance class	Assurance Family	Assurance Components by Composition Assurance Package		
		CAP-A	CAP-B	CAP-C
Composition	ACO_COR	**1**	1	1
	ACO_CTT	**1**	**2**	2
	ACO_DEV	**1**	**2**	**3**
	ACO_REL	**1**	1	**2**
	ACO_VUL	**1**	**2**	**3**
Guidance documents	AGD_OPE	**1**	1	1
	AGD_PRE	**1**	1	1
Life-cycle support	ALC_CMC	**1**	1	1
	ALC_CMS	**2**	2	2
	ALC_DEL			
	ALC_DVS			
	ALC_FLR			
	ALC_LCD			
	ALC_TAT			
Security Target evaluation	ASE_CCL	**1**	1	1
	ASE_ECD	**1**	1	1
	ASE_INT	**1**	1	1
	ASE_OBJ	**1**	**2**	2
	ASE_REQ	**1**	**2**	2
	ASE_SPD		**1**	1
	ASE_TSS	**1**	1	1

Composed Component Interfaces



CAP limitation (1/2)

- The evaluation assurance levels (EALs) and the composition assurance packages (CAPs) have the same structure but apply to different types of TOEs; EALs for component TOEs and CAPs for composed TOEs respectively.
- The Assurance class ACO, newly added in CC v3.1 rev.1, defines information requirements ensuring two or more components for evaluation being integrated in a secure manner; but, it provides no meaningful assurance for the individual component. Hence, it should not be used as augmentation for component TOE evaluations.
- The composed TOE with the highest composition assurance level CAP-C is only able to resistant to attacks by an attacker with extended-basic attack potential.

CAP limitation (2/2)

- For the IT composition products, the CAP is only apply to where the entities of the IT composition products have been evaluated with no further development of either IT entity and the development of additional IT entities or consideration of additional environment entities is not included.
- the CAP is not able to evaluate the IT composition products where the entities of the IT composition products had been evaluated by different evaluation and certification schemes. Should evaluating such a case is required, the evaluation scheme under the CC applied must make provisions for such a evaluation.

TNI/TDI Architectural Consideration

- Partitioned Systems
- Hierarchical Systems

Partitioned Systems (1/2)

- ***Characterization***
  - **A partitioned system is characterized as one that is composed of *cooperating, peer-entity elements***
  - **the various functions to be performed by the system are realized as set of *separate elements*, that communicate via mechanisms such as *parameter-passing* or *message-passing***
  - **each element represents a different function and no subset of the elements constitutes the system described by the requirements**
  - **only the entire collection of elements completely satisfies the system specifications**
  - **Additionally, no element has more privilege or status than any other (i.e., there is no hierarchical ordering of the elements)**

Partitioned Systems (2/2)

- ***Composition Approach***
  - **(a) decompose the policy; allocate policy and functional responsibilities across the system elements;**
  - **(b) develop and evaluate the system elements in accordance with the specifications for each; and**
  - **(c) recompose the system policy; determine that the intended policy is enforced by the combination of system elements.**

Two Types of Networks (1/2)

- ***Unified Network***
  - possess a coherent network architecture and design, and it should be developed with an attention to security requirements, mechanisms, and assurances commensurate with the range of sensitivity of information for which it is to be accredited.
  - accredited as a whole without prior accreditation of their component AIS
  - some of its AIS subsystems are so specialized or dependent on other subsystems of the network for security support that individual accreditation of such subsystems is not possible or meaningful

Two Types of Networks (2/2)

- ***Interconnected Accredited AIS***
  - a network consists of previously accredited AIS
  - a MOA is required between the DAA of each DOD Component AIS and the DAA responsible for the network
  - The network DAA must ensure that interface restrictions and limitations are observed for connections between DOD Component AIS
  - Connections between accredited AIS must be consistent with the mode of operation of each AIS, the specific sensitivity level or range of sensitivity levels for which each AIS is accredited, any additional interface constraints associated with the particular interface device used for the connection, and any other restrictions required by the MOA.

Hierarchical Systems (1/5)

- ***Characterization***
  - **A hierarchically-ordered architecture is characterized by *dependency***
  - **An abstract machine B is "*less primitive*" than an abstract machine A if**
    - **(1) B directly depends on A, or**
    - **(2) a chain of machines exist such that each machine in the chain directly depends on its successor in the chain.**
  - **there is a clear "privilege“ hierarchy**

Hierarchical Systems (2/5)

❑ *TCB Subset*

- ➢ a *TCB subset* enforces a defined policy: it mediates the access of a set of subjects to a set of objects on the basis of stated access control rules
- ➢ a TCB subset is defined such that it may, but need not, include hardware
- ➢ It must be shown to be tamper-resistant, always invoked and small enough to be subject to test and analysis, the completeness of which can be assured.

Hierarchical Systems (3/5)

- *TCB Subset (cont'd)*
  - there is a reference monitor per layer of the system; each layer (subset) is complete with respect to a policy enforced over a set of resources exported by that layer
  - layer m is constrained by the policy enforced by layer m-1; layer m is untrusted relative to layer m-1
  - At the lowest level there is a set of hardware, software, and possibly firmware that implements the TCB

Hierarchical Systems (4/5)

- ***TCB Subset (concluded)***
  - **The next layer of the system treats the previous layer as merely an abstract machine providing resources and exhibiting hardware- like properties**
  - **The concept is recursive; each layer treats the more primitive layers as an abstract machine from which it obtains resources, and by whose policy it is constrained.**
  - **In turn, each layer exports services and resources which are mediated in accordance with the set of access control rules it imposes.**

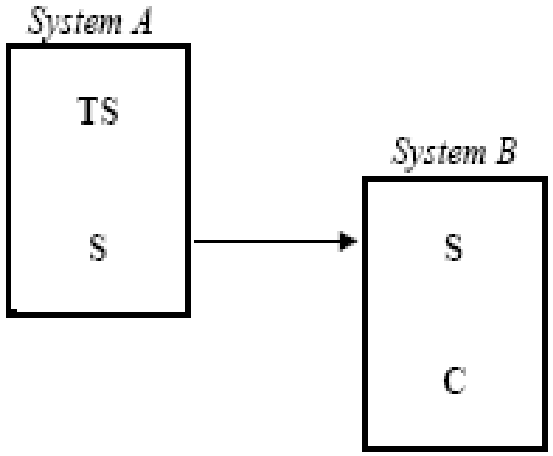
Hierarchical Systems (5/5)

- ***Composition Approach***
  - **(a) decompose the policy; allocate policy responsibilities across the system layers;**
  - **(b) develop and evaluate the system elements in accordance with the specifications for each; and**
  - **(c) recompose the system policy; determine that the intended policy is enforced by the combination of system elements.**

Cascade Vulnerability Problem (1/3)

- Definition: whether the interconnection of secure systems via a secure channel results in a secure distributed system
- It appears in the subset of networks that cannot be treated as a single system due to the large size of the network or the different administrative entities which may lead to different risk assessment methods.
- It appears when independent mutually recognized secure systems are interconnected by secure channels to create a distributed system which is not as secure as its parts

Cascade Vulnerability Problem (2/3)



* Host A is accredited for TS-Top Secret and S-Secret information and all users are cleared to at least the Secret level.
* Host B is accredited for S and C-Confidential and all users are cleared to at least the Confidential level;
* There is a link at level S between the two systems

* To defeat the protection mechanisms of both systems A and B is an easier job than defeating the protection mechanisms of a single system trusted to protect the whole range from TS-level to C-level.
• The network connection has, in essence, created a Trusted Computing Base (TCB) with users cleared to at least the C-level with data on it at the TS-level

Cascade Vulnerability Problem (3/3)

- The network connection has invalidated the risk analysis that accredited the two systems, because such a networked system must have a more secure architecture, a TCB rating of B3, than either rating of the original individual sub-systems TCB (i.e. B1 or B2)

	Maximum Data Sensitivity							
Minimum		U	N	C	S	TS	1C	MC
Clearance	U	C1	B1	B2	B3	*	*	*
or	N	C1	C2	B2	B2	A1	*	*
authorisation	C	C1	C2	C2	B1	B3	A1	*
of	S	C1	C2	C2	C2	B2	B3	A1
System	TS(BI)	C1	C2	C2	C2	C2	B2	B3
Users	TS(SBI)	C1	C2	C2	C2	C2	B1	B2
	1C	C1	C2	C2	C2	C2	C2	B1
	MC	C1	C2	C2	C2	C2	C2	C2

Figure 1. Security Index Matrix for Open Environments [Ref 7]

Suggested Solution to Some of the Composition Problem (1/2)

- In Hierarchical TOEs, make the base component (OS) and the dependent component (DB) as much "*independent*" as possible
  - DB and OS subjects and objects map cleanly
  - DB and OS do not or cannot interfere with each other
- In Peer-to-peer TOEs, designed for minimal *interaction* and clearly discribe remaining interaction (e.g., a PP for applications on the evaluated OS)

Suggested Solution to Some of the Composition Problem (2/2)

- Tackle the easy "composition" problem first
  - Consider the case of taking two or more component TOEs and integrating them for operational use, with no further development.
  - Leave out the consideration of requirements on the non-IT environment or the development of "Glue-ware"
- Start with two TOEs that have been evaluated with composition in mind at low assurance level

Conclusion

- Focus on the composed product evaluation for lower assurance levels from EAL2 to EAL4 individual components
- Preservation of security properties (e.g., information flow control) in composition is a big challenge
- Composition for higher assurance level require more research
- The general composition evaluation is still a significant research problem

References (1/2)

1) Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements, Version 3.1, Revision 1, June 2006.
2) Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 1, June 2006.
3) National Computer Security Center, (1987) *Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria*, Red Book NCSC-TG-005, Library No. S228, 526, V 1, NCSC USA]
4) Wouter Slegers, TNO ITSEF BV “Composition: In search of the holy grail”, ICCC5, Berlin, 2004
5) David Martin CESG UK, “ACO Composition in v3.0”, ICCC6, Tokyo, 2005

References (2/2)

6) H Kurth, and P. Karger, “Suggestion for a Framework for Composite Evaluations”, ICCC5, Berlin, 2004

7) S. Gritzalis, and D. Spinellis, “The Cascade Vulnerability problem: The Detection problem and a Simulated Annealing Approach for its Correction”, *Microprocessors and Microsystems*, 21(10):621-628, April 1998.

8) Mario Tinto, ”The Design and Evaluation of INFOSEC Systems”, The Computer Security Contribution to the Composition Discussion, Library No. S239, 214, June 1992

Thank you

albertjeng@hotmail.com,

doctor_j@jwit.edu.tw