Countermeasure Characterizations
Building Blocks for Designing Secure Information Systems

Herman O. Lubbes
Network Associates, Inc.
Lubbes@tislabs.com

Abstract

The Assurance Working Group (AWG) within the IA Program studied a number of issues relating to the design and analysis of secure systems. A principal element of this work was to understand how to select and integrate countermeasures to form secure systems. It was found that one of the biggest failures of the existing design process was that there was a lack of information about what countermeasures did, how they did it, and how they depended on their operational environment. The Common Criteria documentation provided this information, but the documentation was formal and voluminous. A number of factors led the AWG to adapt an abbreviated format and data description referred to as the Countermeasure Characterization (CMC) containing much of the same information required by the Common Criteria. The countermeasure documentation resulting from the application of CMC data description and format not only supports the system designer, but the thought process necessary to produce it gives the countermeasure developer a better understanding of the environment in which the product must operate.

1. Introduction

Information technologies and their associated computing systems and networks have become a worldwide phenomenon. As these systems have proliferated we have seen increasing instances of hostile attacks resulting in loss of data, services, time, money and confidence in these vital systems. This paper addresses ways to design systems that increase our confidence that they are resilient to hostile attacks and contain minimal vulnerabilities due to flaws in their design.

The DARPA Information Assurance Program (IA) sought to develop countermeasures against hostile attacks. Under several contracts, individual Principal Investigators (PIs) attempted to develop countermeasures against a variety of threats. The Assurance Working Group (AWG), an informal working group composed of personnel from NSA, Mitre, NRL and NAI[1], was tasked to: 1) provide a way to quickly communicate the objectives and accomplishments of each PI to the other PIs; and 2) to provide assistance to the PIs in their efforts to integrate their individual technologies into complete secure information systems. This paper discusses some of the significant observations and findings arising during the task period, (September, 99 to September, 00).

First, we look at prior and current methodologies for designing systems with known assurance levels. Next, we discuss the current state of practice as reflected by Common Criteria (CC)[1] evaluation methodologies, and a prospective improvement we call Countermeasure Characterization (CMC). Finally, we postulate a number of benefits that could accrue from further development of the CMC.

2. Background

In 1984, Carl Landwehr, Connie Heitmeyer, and John McLean published "A Security Model for Military Message Systems" [2]. This formal model consisted of three parts:

- Definitions
- Assertions
- Assumptions

Assertions are predicates to be proven or demonstrated to be correct for a component of a system, given a set of assumptions about other components or security disciplines that establish the environment of the component making the assertions. The component developer making the assertions to be proven has no control over the correctness of the assumptions made about the other elements or disciplines. The other disciplines include physical security, operational

[1] DARPA Prime Contract #F30602-98-C-0012, Purchase Order #501298

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	6/14/2001	Research Paper 6/14/2001

4. TITLE AND SUBTITLE
Countermeasure Characterizations Building Blocks for Designing Secure Information Systems

5. FUNDING NUMBERS

6. AUTHOR(S)
Lubbes, Herman O.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

DARPA

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

IATAC
3190 Fairview Park Drive
Falls Church, VA 22042

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; Distribution unlimited

12b. DISTRIBUTION CODE
A

13. ABSTRACT ***(Maximum 200 Words)***

The Assurance Working Group (AWG) within the IA Program studied a number of issues relating to the design and analysis of secure systems. A principal element of this work was to understand how to select and integrate countermeasures to form secure systems. It was found that one of the biggest failures of the existing design process was that there was a lack of information about what countermeasures did, how they did it, and how they depended on their operational environment. The Common Criteria documentation provided this information, but the documentation was formal and voluminous. A number of

14. SUBJECT TERMS
IATAC Collection, information assurance, Common Criteria

15. NUMBER OF PAGES
13

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

security, and personnel security. The most significant aspect of this work was that it clearly illustrated that a component's correct operation is dependent on the correct operation of its environment. It also illustrated the need to clearly state all the assumptions that must be true about a countermeasure's environment for the countermeasure to function correctly. Efforts in this field using the ideas of Landwehr et al. are described in References [3] and [4].

2.1 Technical Base Line

Some of the most difficult issues in the design of secure information systems are:

- Ability to predict the assurance level of a system composed of components that have different:
  – levels of assurance;
  – documentation standards; and
  – evidence that its assurance level is valid.
- Understanding how to add components, including countermeasures, to a system without adding exploitable flaws.
- Determining the correct selection of countermeasures for a system and its application.

The AWG has been studying aspects of these various issues since the beginning of the Information Assurance program in DARPA. The AWG focus has been to identify effective ways of describing essential and distinctive countermeasure features and dependencies. This information is required by system designers and integrators to enable selection and integration of these countermeasures into secure systems. This information and the process of generating it are referred to as Countermeasure Characterization (CMC). One way of stating the required relationships is that "claims" are the countermeasures output and "assumptions" are the inputs necessary to validate the CMC's operations. In order to be widely used and useful the CMC must be easy to create, easy to read and easy to understand. This paper describes the CMC and its development. It also describes future work that could expand the use of the CMC as a building block for secure systems.

2.2 Common Criteria (ISO 15408)

The Common Criteria (CC) is an international standard for evaluating the security of IT products based on existing US, European, and Canadian criteria for IT security evaluation. In October 1998, Canada, France, Germany, United Kingdom, and the United States signed a Mutual Recognition Arrangement (MRA) for Common Criteria based evaluations. In May 2000 Australia, New Zealand, Finland, Greece, Italy, Netherlands, Norway and Spain joined with the original group in signing a new MRA. A CC evaluation awarded in one country is recognized by all member countries.

The CC approach allows prospective users to obtain an impartial assessment of an IT product by an independent laboratory. This security evaluation includes an analysis of the IT product and the testing of the product for conformance to a set of security requirements. A Protection Profile (PP) document defines the security requirements for the IT component to be procured. A second document called a Security Target (ST) is developed for the component proposed for meeting the PP requirements. The specific IT product being evaluated is referred to as the Target of Evaluation (TOE). The ST illustrates how the TOE meets the PP requirements.

While the CC enjoys the advantage of universal acceptance with regard to evaluating component security, a number of problems arise when using it as a system level tool. The CC provides little or no effective support for designing or analyzing composite systems; rather it is successful only when applied to individual components or products.

2.3 Visual Network Rating Model

NRL is developing a tool called the Visual Network Rating Model (VNRM) that makes use of the concepts developed by Landwehr et al. VNRM is used to draw assurance maps using claim trees. It is a graphical language used to portray much the same information as the CMC. NRL is using it to map assurance arguments to various parts of components so that one can see how individual arguments work together to support the component assurance argument. The tool also helps one to identify gaps in the argument string. These gaps suggest vulnerabilities that may be exploitable. VNRM appears to be an excellent tool for supporting the CMC process. VNRM has the ability to clearly illustrate countermeasure dependencies on their environment and any inconsistencies in these dependencies. VNRM claim trees are flexible. They can be done at a high level of abstraction or in significant detail depending on need, giving VNRM the potential to be a powerful tool for design and verification of secure systems. For more detailed information on claim trees using the VNRM, see Reference [5]. Other work using similar tools is described in Reference [6].

NRL is developing the Network Pump (NP) for the purpose of providing reliable communications from a system-high network operated at a low security level to a system-high network operated at a high security level, while controlling the band-width of the potential covert timing channel between high and low to an arbitrarily low level. As part of the NP development efforts, NRL

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

has developed both a CC ST [7] and a Network Pump Assurance Argument [8] using VNRM. These documents are both partial drafts that have not had peer review. The author of these two documents did not plan to ensure consistency between the documents except that they both would illustrate that the NP satisfied its objectives. With the author's permission, the two documents and the two methodologies used to develop them were compared. The purpose of the comparison was to discover the differences in the documentation resulting from the two methods and to partially identify the pros and cons of each tool.

A complete comparison could not be made because the VNRM document is less complete with regard to identifying threats and therefore identifies fewer claims to counter threats. The comparison was complete enough, however, to determine that there is a one-to-one comparison between the information required by both CC and VNRM and the way this information is used. Both approaches show how claims made by the countermeasure, in conjunction with assumptions made about other parts of the system (the countermeasure's environment), combine to counter threats to the system and/or the countermeasure itself. To further verify this conclusion, a document describing the effectiveness of OO-DTE (written in the language of the CC) [9] was successfully translated into claim trees using VNRM [10]. OO-DTE is more fully described in Reference [11].

3. Countermeasure Characterization

The AWG studied both the Common Criteria and VNRM during the process of formulating Countermeasure Characterizations. Because it was still under development, VNRM did not provide a suitable basis for CMC. Two or more persons, each equally versed in a particular countermeasure technology and VNRM could easily develop different claim trees that are both technically correct. The tool is complex and did not handle repeated structures very well during the experiments. There is no large-scale training or technical support available to developers or vendors and currently there is little, if any, motivation for vendors or other developers to take on the extra burden of developing claim trees. The Common Criteria documents are quite voluminous and written in a semi-formal rigid language that impedes prompt analysis. However, if this standard becomes widely used it is likely that vendors will use Security Targets (ST) for documenting their products as required by the Common Criteria. Starting with a well-written ST or PP one can easily create a CMC.

Based on the above factors, particularly the fact that the CC is a government standard having the desired information content, the AWG decided to base the CMC data content on the CC. However, in order for the CMC to be useful, it had to be lightweight and informative, yet easy to produce.

CMCs are "essentials only" versions of CC documents. They are only about ten pages in length and describe the technology without using the CC's complex language. CMC documents the essential and distinctive countermeasure features and dependencies needed to support the selection and integration of countermeasures into secure systems. This is done with a fair amount of structure, but unlike the CC, CMC has enough flexibility to permit advertising security features from a competitive marketing point of view. Using the CC, it is difficult for vendors or developers to convey unique attributes of their work or products. CMCs have been found to be useful by developers of security technology prototypes who have used the technique to document their efforts.

We do not propose a replacement for the CC or any part of it rather, we see CMC as a small, but important tool in a large set of tools needed to address the integration issues. The CMC is more responsive to the AWG tasking than the CC because it is much shorter, easier to write (in ones own language instead of the formal CC language), and easier to read for the same reason.

3.1 CMC Format

The following information content description and format was developed to define the countermeasure characterizations. The approximate page counts is as follows: *The information in italics was added for the purposes of this paper and is not part of the CMC description.*

Basic description of the technology [~1 page] (for selecting a technology)

What does the technology or product do? Provide a succinct description of the technology or product and its functionality from a system context.

→ Sample questions concerning technology or product: Does it provide operating system functions? Does it provide networking functions? Does it provide an interface among two or more networks? Does it route messages? What specific hardware and software are involved? What are its limitations? The answers to these questions may depend on how the technology or product is used in a system context. Some technologies or products may perform only security related functions.

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

Security problem addressed [~1 page]

What security problem does it solve? Provide a succinct description of the security functionality. If the technology or product provides only security functionality than this section and section 1, may overlap.

→ Sample questions concerning the technology: What threats does it counter? Does it provide identification and authentication (I&A)? Does it provide access control? Does it perform intrusion detection? What are its security limitations?

These first two sections provide the first level of information needed to select a countermeasure.

Assumptions [~1 page]

What assumptions are needed about the *environment* of the technology or product and their *intended use*? Provide in list form assumptions that must be true about the environment for correct functioning and security of the technology or product.

→ Further explanation: Assumptions represent information about other components or disciplines (physical, operational, or personnel security) over which the subject technology has no control that must be true in order for the subject technology to function correctly. For example, if a guard does not check the format of its input data, then it must assume that the data format is correct even though it likely comes from an untrusted system. An access control system might assume that the files it uses to make access decisions are protected by its host operating system. For almost all security mechanisms proper configuration by authorized, competent, properly trained system administrators is assumed. Assumptions describe the dependencies of the countermeasure on the rest of the system and its environment. Assumptions that are not true represent vulnerabilities or exploitable weaknesses. Invalidated or unsupported assumptions represent likely vulnerabilities. If the input data to the guard is not formatted properly, then the guard will likely make an incorrect release decision. If the operating system does not protect the files used to make access control decisions, then unauthorized personnel can modify the files. Unless a countermeasure is properly administered, it is vulnerable. This section should describe the assumptions that the technology makes about the components with which it interfaces and the assumptions it makes about its operational environment including the physical, operational, and personnel security disciplines.

This may be the most important information in the CMC. It is usually the most poorly documented yet it is essential to support the composition of countermeasures into secure systems. The process of selecting a countermeasure may be supported by the information on assumptions giving the designer choices of countermeasures that are more or less dependent on other parts of the system.

Threats and attacks [~2 pages]

What threats and attacks does it counter? Provide a list of general and specific threats and attacks countered by the technology. Provide an additional list of vulnerabilities, threats, and attacks to which the technology, or countermeasure itself, is vulnerable. This list accounts for residual risks.

→ Further explanation: The list of general threats and attacks can be derived from known sources like the ones listed in Reference [12].

Security objectives [~2 pages]

What are its security objectives? Provide a list of the security objectives for the technology and its intended environment.

→ Further explanation: Security objectives are actions that the technology must accomplish in order to counter threats and attacks and/or support any identified organizational security policies. These actions must be considered in light of the assumptions made about other related components and the environment. For example, the OO-DTE technology assumes that the underlying operating system will protect the integrity of the local copy of the policy, role-authorizations database, and executable files. One of the OO-DTE objectives is that it must use the role definition to prevent users from gaining access to and performing operations on its resources/objects, unless the users have been granted access by the resource/object owner, or they have been assigned to a role (by an authorized user) that permits those operations. This objective can only be performed correctly if the role-authorization database's integrity is maintained by the underlying operating system. In the rationale item a mapping is made to illustrate how the individual objectives act to counter the threats. In some cases only one objective is necessary, in other cases, several objectives may be necessary to counter a specific threat.

Rationale [~2 pages]

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

The rationale explains how the objectives together with the assumptions counter the threats and attacks addressed by the technology. In rough symbolic form

Objectives + Assumptions > Threats and Attacks - Residual Risks

where '>' means 'counters'. For the rationale provide a table of threats and attacks versus objectives and assumptions with a descriptive set of sentences or argument indicating why the identified security objectives of the technology together with the assumptions counter the identified threats.

A table offers a useful way to map the threats to the objectives and assumptions addressing them.

	Threat 1	Threat 2	Threat 3
Object. 1	X		
Object. 2		X	X
Object. 3			X
Assum. 1	X		
Assum. 2			X
Assum. 3		X	

Sample Table illustrating relationships between Threats, Objectives, and Assumptions.

The last three sections provide the detailed information about how the countermeasure works in conjunction with the assumptions to counter specific threats. This information can also be used in analyzing a system to understand the impact of making assumptions that can't be verified.

The CMC format and content draw much from the CC specification for a PP and where appropriate an ST. The PP Introduction and the TOE (Target of Evaluation) Description sections of the PP are equivalent to the first two sections of the CMC (Basic Description of the Technology and Security Problem Addressed). The content of the PP is illustrated in Figure 3.1-1. The shaded area in Figure 3.1-1 shows the information that we didn't use from the PP. The AWG decided that the detailed information in the PP not included in the CMC was not central to the purpose of the CMC. This was a judgment call that may need to revised based on experience.

We removed references to evaluation because it was not consistent with the purpose of the CMC. We kept the other content of these parts of the PP because it provides for a useful description of the countermeasure. The information required in the TOE security environment of the PP is the same as that required under the headings of Assumptions and Threats and Attacks in the CMC. Threat discussions further enhance the understanding of the purpose of the countermeasure being characterized. The discussion of Assumptions in both the PP and the CMC is very important to proper integration of countermeasures into systems.

The requirements for this information in the documentation described by this paper may be unique. The requirement for discussion of the security objectives is basically the same for both the CC documents and the CMC. Objectives are the actions that the countermeasure must take to counter the threats. The PP requires that the objectives be supported by selections from a large library of formally written functional specifications. Both the CMC and the PP require written rationale illustrating how the threats are countered by a combination of objectives and their associated assumptions. The PP requires additional written rationale to illustrate how the functional specifications support the objectives.

3.2 CMC Experiment

The AWG needed a way to obtain experience with the CMCs in order to validate its decisions concerning their content and format. Since the PIs were the initial developers and users of the first CMCs they would be able to provide valuable feedback on issues such as the difficulty in developing the CMC, its usefulness to the developer of countermeasure technology, and its usefulness to a consumer of that technology. By providing minimal training in the development of the CMCs we could get an idea of the difficulty involved in their development. Because there was a wide variety of technologies being developed by the PIs we could also get a feel for the flexibility of the CMC. Each of the IA program PIs was directed by the DARPA program manager to develop a CMC for their technology effort and a one day workshop was organized to provide assistance to the PIs in generating the CMCs.

A week before the workshop, we had a chance to work with one of the IDIP developers who had shown a continuing interest in what we were trying to accomplish [13]. Several interesting issues were raised during these discussions. We recognized that we would want to integrate the CMCs into systems. In order to fit together CMCs had to be described at the same level of abstraction. How would we know the correct level of abstraction until we started putting the CMCs together? The IDIP PI raised a related issue. The IDIP program's objective was to develop a protocol to protect and communicate intrusion detection sensor data. IDIP was only part of a countermeasure system. The question was: Would it be useful to create a CMC for the IDIP function by itself? The consensus was that it only made

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

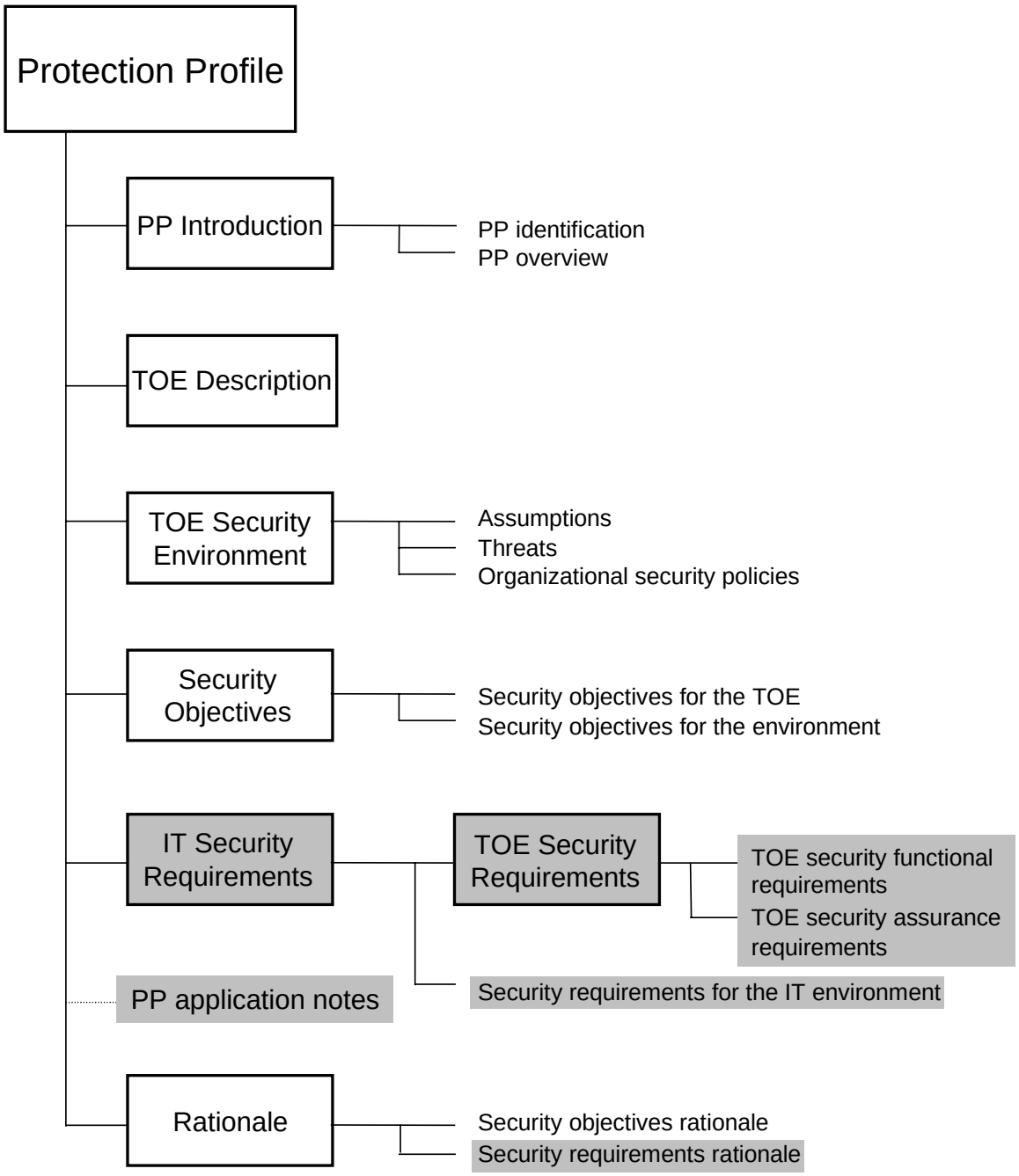

Figure 3.1-1 Protection Profile

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

sense to characterize the IDIP protocol as part of a countermeasure system.

During the course of the CMC experiments some 13 CMC's covering a broad range of technologies were generated. A listing of these documents is provided in Appendix A. The feedback received from the PIs who generated the CMCs convinced us that the CMC was a good idea. Most of them confirmed that there was value to the developer in using the CMC format and content definition because it caused them to think carefully about what they were trying to accomplish and the assumptions they were making about the environment in which their technology would operate. Based on their requirement to integrate their developmental technology with that of other PIs, they concluded that the CMC provided useful information to support this integration. The majority of the PIs said that although the writing of the CMC was reasonably simple, a lot of thought had to go into its generation.

A particularly significant reaction from a participant in the workshop was that the IA community should aim higher on the CMCs effort. He visualized the CMCs being organized to provide reference material like a Physicians' Desk Reference (PDR) for IA tools and technologies. The reference would contain:

- References
- Source Code
- Reviews with attribution
- Links to other relevant topics
- Comprehensive list of dependencies and other assumptions for each technology

3.3 Observations

The author believes the AWG has designed an approach for describing countermeasures that maximizes the information communicated about their purpose, functions, environmental dependencies, and concept of operation while minimizing the amount of documentation. The act of generating the CMC causes designers to examine their work in a structured way that assists in the identification of vulnerabilities that might otherwise not be identified until the countermeasure is operational. The CMC assists the system integrator by identifying assumptions made by the countermeasure that must be satisfied by other system information technology components or functions of other security disciplines (physical, operational, or personnel). As people work with the CMC ways to improve it will no doubt be identified. A suggestion to add an additional item to the CMC format to separately describe residual risks has already been made.

4. Continuing Efforts

By combining the CMCs with the VNRM claim tree tool one can postulate support tools for the composition of secure systems. Claim trees have been used primarily to map assurance arguments to system components, but they have significant potential as aids to composing secure systems. Security relevant system components and subsystems can be defined in the form of CMCs. Claim trees illustrate two characteristics that form a system's component's or subsystem's security interface with other security relevant components and other system security disciplines (e.g., physical, procedural, and personnel) [14]. The first characteristic is the set of claims made (with some level of assurance) about the subject component's security functionality (e.g., controls access to resources/objects based on user roles). In VNRM, it can be shown how claims at a higher level of abstraction are supported by claims at lower levels of abstraction. The second characteristic is a set of assumptions about other IT components or other disciplines. These assumptions must be true for the claims to be true but the assumptions are outside the control of the component making the claims. A simple but useful way of looking at these relationships is that claims are the component's output and the assumptions are its inputs. The input the component expects to see are assumptions about other IT components. Research needs to be done to create and validate a set of rules for composing secure systems from CMCs. One such rule may be that to compose a secure system: all of the assumptions upon which the claims of one component depend must be supported by claims made by other IA components or disciplines. Failure to satisfy this composition rule creates a vulnerability.

The composition process assumes the existence of a library of CMCs from which the needed security functionality described by the CMC will be integrated into a system. The format and contents of the CMC library must be compatible with the claim tree structure and at the right level of abstraction to compose correctly. Other capabilities that are needed include search algorithms to identify the needed countermeasure and a set of composition rules.

4.1 CMC Evolution

To design a secure system one begins with the most general security requirement. As the refinement of the requirement proceeds, through lower levels of abstraction using VNRM necessary security functions are identified. At this point in the process one refers to

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

the library of CMCs. Based on the information in the CMCs, the designer identifies potential countermeasures able to provide the required security functions. The basis for this selection may be that one countermeasure has significantly stronger evidence that its claims are true. Another basis may be that one countermeasure requires fewer assumptions about its operating environment that must be validated by claims of other components. In the process of composing a secure system, the claim tree must be refined until one can be assured that the composition rule has been satisfied or one can determine that the residual risks are not significant. During the design process, claim trees are used in two ways. The first is to break down the high-level security requirements into security functions that are represented by CMCs. The second is to assist in verifying that the composition rules have been followed. During the refinement process, CMCs are substituted for the pieces of the claim trees that describe specific security functions or countermeasures. This action represents a design decision at some level of abstraction depending upon the level of detail in the CMC.

The author believes that in addition to providing a way to convey information about countermeasures and evolving countermeasure technology, CMCs are critical components of a methodology for designing systems with known levels of assurance based on mission need. Figure 4.1-1 is a conceptual diagram of the methodology. The upper part of the sketch represents a generic set of claim trees. The top level represents a claim that the system meets the most abstract system security requirement. As the claim tree branches out, the security requirements become less abstract and a security architecture is defined. As the architecture becomes more refined, requirements for components with specific security functionality are identified. CMCs that document security components that match the identified functionality can be substituted for branches of the claim tree that would otherwise be needed to document them. The right side of the diagram is a conceptual illustration of how the component claims support the top-level system security requirement via the claim trees.

4.2 Composition Process Limitations

In order to compose a secure system from secure components, all of the assumptions upon which the claims of one component depend must be supported by a claim that can be made about another IT component or IA discipline. The process limitations of this approach for obtaining information assurance in an integrated system fall into three classes or types: The process can break down for the following reasons:

1) The assumptions may be incorrect, incomplete, or at the wrong level of abstraction to appropriately match up with the supporting claims.
2) The claims may be incorrect, incomplete, or at the wrong level of abstraction to appropriately match up with the assumptions that they are intended to support.
3) Some of the IT components in the system may not have a security interface described in terms of claims and assumptions. For some of these, it may be possible to develop the necessary security interface. In other cases, for example most COTS products or proprietary software, it may not be feasible to describe these security interfaces, or the underlying assurance that these descriptions are correct and complete may be nonexistent or very low.

The composition assurance argument is based upon a form of chaining assurances. Components are composed with the assurance that the assumptions for one component are met by depending upon the claims of another. This results in an assurance argument from the input of the first component or components in the chain through to the outputs of the last component/components in the chain. Even if the assurance arguments are sufficient for all possible chains of components in the system, this is still incomplete. These assurance arguments are for a view of the system as a composition of components at a given level of abstraction. The method does allow for drilling down to arbitrarily low levels of detail; however, system assurance comes from maintaining such assurance arguments consistently across all levels of abstraction. To do this completely, even if such were possible, may require enormous expenditure of resources. In addition, the resulting assurances may not justify such expenditure.

4.3 Further Research

Work that needs to be done to extend the usefulness of the CMC includes the following:

- Develop a cataloguing system to identify countermeasures that meet specific criteria.
- Devise a way to make the CMCs broadly available such as a web site.
- Develop fuller understanding of the abstraction issue and how to deal with it in order to use the individual CMCs as building blocks in an integrated system.

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

- Develop a process for combining the strength of claim trees and CMCs to develop and analyze secure systems.

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

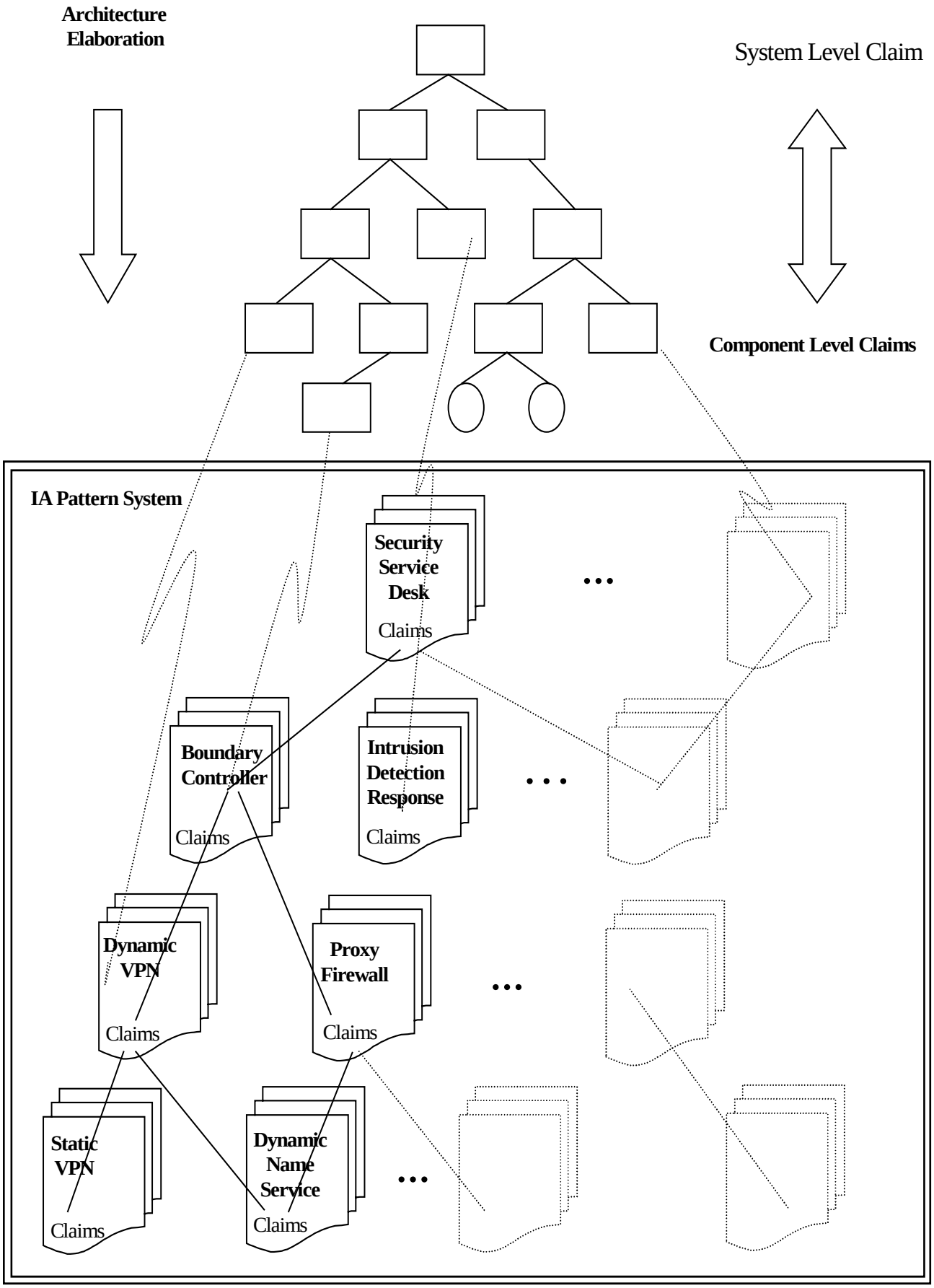

Figure 4.1-1 System Integration Assurance Map

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

As pointed out by the PIs, there needs to be a way to quickly and easily distribute the CMC information. One suggestion is a web site with broad access rights. That may be a reasonable solution to the distribution problem but there still exists a need for coding and indexing schemes and search algorithms. Further research is needed to fully understand how to use the CMCs or similar structures like patterns to compose secure systems. Areas that require further research include:

- Rules for composing different types of CMCs
- Solution to problem of inconsistent levels of abstraction. One possibility is to have CMCs at multiple levels of abstraction.
- Methods and rules for combining CMCs and claim trees
- Rules for the composition of secure systems
- The effect of security aware and unaware applications on the composition process
- Addressing the issues raised in Section 4.2

5. Summary

The author believes the AWG has suggested a data description and format that should be used to describe essential and distinctive countermeasure features and dependencies. This data description and format is the CMC. Its primary purpose is to communicate information about countermeasure technology and products among researchers, product vendors, and systems integrators. It conveys a significant amount of information in a document approximately 10 pages in length but requires careful thought to produce. This thought process has been found useful to the developers who have gained a more complete understanding of the problem they are attempting to solve and the environment in which they are trying to solve it. It has helped uncover hidden assumptions leading to unexpected vulnerabilities. The author believes that there is significant potential for the CMC or patterns [15] containing similar information as building blocks in tools to aid developers in the composition and integration of secure systems. There is probably no one set of tools that can completely address all the issues identified at the beginning of this paper. However, the AWG believes that using the CMCs in conjunction with the VNRM claim trees can provide a better understanding of many of the issues and answers to some. As several PIs pointed out, it is critical that CMCs be easily available or they're not useful to the system integrator. A potential way to address this issue is for the AWG or a member of the AWG to sponsor a web site that would hold and distribute CMCs generated both by government sponsored researchers and vendors of COTS products.

Acknowledgements

The author wishes to thank all of the AWG members who helped formulate the ideas in this paper by offering their comments, suggestions, and debate. Special thanks are extended to Andy Moore, who contributed significantly to the ideas behind CMC and its potential extensions and to Lee Benzinger for much of the information presented in Section 4.2, that helped us understand that the composition problem is more complex than it first appears. Special thanks are also extended to Diann Vechery for her help with earlier versions of this paper.

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

References

[1] The Common Criteria for Information Technology Security Evaluation (CC) version 2.1: ISO International Standard 15408, 19 September 2000.

[2] C. Landwehr, C. Heitmeyer, and J. McLean, "A security model for military message systems", ACM Transactions on Computer Systems, vol.2,pp198-222, August 1984

[3] Pane, C.N., J.N. Froscher, C.N. Landwehr, "Toward a Comprehensive INFOSEC Certification Methodology", Proc. 16th National Computer Security Conference, pp. 165-172, Baltimore, MD, September 1993.

[4] Herman O. Lubbes, Distinguished Lecturer, "Computer Security, A Personal View", Applications Conference, Orlando, FL, December 1994.

[5] Moore, Andrew and Beth Strohmayer, "Visual NRM User's Manual: Tools for Applying the Network Rating Methodology", NRL Formal Report NRL/FR/5540-00-9950, 31 May 2000.

[6] D.M. Kienzle and W.A.Wulf, "A Practical Approach to Security Assessment", Proc. New Security Paradigms Workshop, Langdale, Cumbria, UK, September 1997

[7] Moore, Andrew P., "Network Pump (NP) Security Target", NRL Memorandum Report 5540-00-8459, 29 May 2000.

[8] Naval Research Laboratory, "Network Pump Assurance Argument (Draft)", NRL Technical Memorandum 5540-135A:apm.

[9] Peter Dinsmore, "Security Effectiveness of OO-DTE", NAI Labs Report # 0770, 6/28/99.

[10] Herman O. Lubbes, Diann Vechery, "OO-DTE MPOG Security Effectiveness Claim Trees", an application of the network rating methodology, NAI Labs Report #00-026, 9/14/99.

[11] Daniel F. Sterne, Greg W. Talley, C. Durward McDonell, David L. Sherman, Pierre X. Pasturel, E. John Sebes, "Scalable Access Control for Distributed Object Systems", Proceedings of the 8th Usenix Security Symposium, August 1999.

[12] Defense Information Assurance Red Team Methodology, MP 98B0000018, April 1998.

[13] Dan Schnackenberg, Kelly Djahandari, Dan Sterne, "Infrastructure for Intrusion Detection and Response", Proc. DARPA Information Survivability Conference and Exposition (DISCEX)2000, Hilton Head, SC, Jan. 2000.

[14] DoD, "The Network Rating Methodology: a Framework for Assessing Network Security", September 1997.

[15] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns, Elements of Reusable Object-Oriented Software, Addison Wesley Publishing Co., ISBN 0-201-63361-2, 1995.

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE

Appendix A

Documents for Countermeasure Characterization Workshop, 18 November 1999

OO-DTE Countermeasure Characterization
Jan Filsinger 07/31/2000

ARGuE Countermeasure Characterization
Dale Johnson 07/31/2000

Differential Filters Countermeasure Characterization
Tom Markham 07/31/2000

Policy Enforcing Network Interface Card Countermeasure Characterization
07/31/2000

MPOG Countermeasure Characterization
David Sames/Robert Lyda 07/31/2000

DVPN Countermeasure Characterization
Darrell Kindred 07/31/2000

CCS Countermeasure Characterization
David Levin 07/31/2000

Napoleon Countermeasure Characterization
Dick O'Brien/Dan Tomsen 07/31/2000

Principal Middle Manager Characterization
07/31/2000

SMARTS Countermeasure Characterization
Rich Feiertag 07/31/2000

IDIP Countermeasures Characterization
D.Schnackenberg 01/10/2000

NRL Network Pump Countermeasure Characterization
Andy Moore 11/15/1999

ISBN 0-7695-1212-7/01 $10.00 (C) 2001 IEEE