THE HANDBOOK FOR THE COMPUTER SECURITY CERTIFICATION OF TRUSTED SYSTEMS*

Judith N. Froscher
Charles N. Payne, Jr.
Code 5542
Naval Research Laboratory
Washington, D.C.

October 12, 1992

Abstract

The Navy has designated the Naval Research Laboratory (NRL) as its Center for Computer Security Research and Evaluation. NRL is actively developing a Navy capability to certify trusted systems. This paper describes the NRL effort to understand assurance, certification, and trusted system certification criteria through the production of the Handbook for the Computer Security Certification of Trusted Systems. Through this effort, NRL hopes to discover new and more efficient ways of satisfying the assurance requirement for a high assurance system.

1 Introduction

In the past few years, DoD policy makers have begun to view computer security as an enabling technology, which will allow the Services to share classified information securely and legitimately. Several factors motivated this change in policy. The computer security community, primarily through the leadership of the National Computer Security Center, convinced industry that there is a real need for computer security products. Industry responded with the development of new products that are beginning to populate the Evaluated Products List (EPL), especially at the lower trust classes of the *Trusted Computer System Evaluation Criteria* (TCSEC) [1]. Widely publicized penetrations of Government computers also encouraged widespread interest in computer security. The computer security community is challenged to produce systems that protect classified information, that satisfy critical requirements, and that provide assurance to their users that they are trustworthy.

The development of a software engineering methodology for trusted systems has been a major research goal at NRL for several years. The objective of this project is to define a system development process whose enactment provides assurance that the computer system satisfies its critical, security requirements. Given such a trusted development process, certification that the requirements are satisfied becomes a technical audit of the development process. A requirement for this endeavor is a thorough understanding of the assurance required for certification.

The TCSEC is the primary security certification criteria for the US. To better understand the assurance required by the TCSEC, NRL is developing a computer security certification handbook for trusted systems. The focus of this project is the B3 class of the TCSEC primarily because our objective is to investigate the different kinds of assurance provided by different development strategies, which include both rigorous design and engineering approaches as well as formal methods. This handbook will serve other purposes as well. NRL's goals are to develop a Navy capability to certify trusted systems, to document an evaluation methodology and train Navy evaluators, and to comply with SECNAVINST 5239.2, which designates NRL as the Navy Center for Computer Security Research and Evaluation. This short paper describes the NRL effort to understand assurance, certification, and trusted system certification criteria through the production of a Handbook for the Computer Security Certification of Trusted Systems.

*Presented at MILCOM '92, San Diego CA, October 1992

2 The Trusted System

The term, *trusted system*, has a specific meaning in the Handbook. NRL has adopted the definitions of *product* and *system* as stated in the European community's *Information Technology Security Evaluation Criteria* (ITSEC) [2]. According to the ITSEC, a *system* is a specific installation "with a particular purpose and a known operational environment". A *product*, on the other hand, is "a hardware and/or software package that can be bought off the shelf and incorporated into a variety of systems". This distinction between a product and a system is only implicit in the TCSEC. A *trusted system*, then, is a *system* that has been certified against some trust criteria.

The characteristics and requirements of a trusted system's end–users are well known, and threats to a trusted system's security can be determined with some certainty. The security requirements that it must enforce are unique interpretations of a national security policy. If the trusted system is based on a trusted product, the person deploying the trusted system must ensure that the assumptions of the trusted product are valid for the operating environment. It may be necessary to develop additional trusted code to enforce environment–specific security requirements.

The ITSEC notes that for the sake of consistency, the same trust criteria should be applied to both trusted systems and trusted products. The TCSEC, on the other hand, asserts that while its assurance requirements can be applied "to the full range of computing environments", its security feature requirements are targetted primarily at "information processing systems employing general–purpose operating systems that are distinct from the application programs being supported", i.e., trusted products. NRL assumes that the TCSEC can be extended to trusted systems. The certification evidence that is assessed for a trusted product must be assessed for a trusted system also.

3 Computer Security Certification

Certification contributes significantly to the demonstration that a system is trustworthy. Certification supports the accreditation decision to allow the system to process classified information in an operational environment. Trusted system certification[1] comprises several technical and procedural certifications, including a technical computer security certification of the system's security features. Computer security certification is the independent technical evaluation of a trusted system to determine whether the system satisfies a set of critical operational and assurance requirements, e.g., the TCSEC. The outcome of the computer security certification influences the criteria for other system certifications, such as administrative and personnel security. If the protection features of the computer system are deficient in any way, other protection measures must be employed to protect the information and processes controlled by the system.

The computer security certification of a trusted product and the computer security certification of a trusted system may be based on the same criteria, but the effort involved is significantly different. Trusted product evaluations can take three to four years. The vendor accepts this rigorous process because he knows that his product can be sold to many different customers. Trusted systems are for a single customer. The customer usually cannot tolerate a long delay in the delivery and/or operational use of the trusted system. Alternate certification approaches must be explored to reduce this delay.

One possible approach, which is similar to the trusted product evaluation process, is to provide security engineering expertise to the program manager and to the developer during system construction and then to assign an evaluation team to assess the completed system. The security engineering support team must be different from the evaluation team in order to preserve the independence of the evaluation. Although this approach is straightforward and a delivered system is likely, it does not allow opportunity to change course or correct mistakes during development. The delivered system may not be certified at the required TCSEC class and may require redevelopment to satisfy its security requirements. In general, performing a technical evaluation after the system is completed can significantly delay operational use of the system.

Another approach is to structure the certification as an independent security verification and validation process. Since the TCSEC describes evidence for a completed trusted product, the challenge is to determine what certification evidence is needed at each milestone of the development process in order to determine that the resulting system can be certified. Certifiability is defined as the contractual acceptance of the certification evidence for a particular milestone, and it is assessed at each milestone. This approach is success ori-

[1] The TCSEC calls this a *certification evaluation*.

ented: if a system is delivered, it can be certified. Unfortunately, the system may never be delivered, particularly if the system requirements are not well understood. This approach can also lengthen the development time significantly and threaten the independence of the evaluation team.

Our goal is to define a trusted system development process that employs the assurance criteria required for certification to support development. The development environment would be based on a definition of trusted configuration management. Only authorized members of the development team could make changes to documentation and code; only authorized members of the certification team could generate certification assessments of certain evidence. Most of the certification process would evolve into an audit of the assurance processes performed during the trusted system's development. Throughout the development lifecycle, certifiability would be continuously assessed and ensured. The delivered trusted system would be certifiable at the target class and the independence of the evaluation would be preserved. The trusted development environment would remain available to the system maintainers and would ensure that the only changes permitted to the system are those that result in a certifiable system.

Since the definition of this trusted system development process relies heavily on an understanding of assurance and the contributions of the TCSEC certification evidence thereof, the next section explores the meaning of assurance and identifies some of the special issues that must be considered for trusted systems.

4 Assurance for Trusted Systems

Assurance is the confidence gained from compelling evidence that a system satisfies its critical requirements. Assurance about critical system behavior results from many factors, including the unambiguous specification of critical requirements, formal specification and proof techniques, visibility into the system design and development process, well structured software architectures, rigorous schemes for demonstrating correspondence between a system and its requirements, independence among the software building blocks, understandable documentation, testing, independent evaluation, and the credentials of the system developers and evaluators. All of these activities contribute to the assurance argument for a trusted system; no single activity provides evidence that convinces users conclusively that the system behaves as expected in the context of its critical requirements.

The previous section noted that while certifications for trusted products and trusted systems may be based on the same criteria, the effort involved is very different. The difference is borne from the assurance argument that must be constructed to support the certification. The trusted system's assurance argument consumes greater development resources than a trusted product's because the trusted system's security policy is more comprehensive than the trusted product's. For example, the trusted system's security policy may name individual users and identify unique relationships between them. The same is true of the objects that must be protected by the system. In addition, the trusted system must enforce unique interpretations of a security policy. All of this means that the assurance argument for a trusted system can be very complex.

The complexity of a trusted system's assurance argument affects the content of the certification evidence. For example, a trusted product enforces a relatively simple, well–understood security policy, so its formal model of the security policy (FMSP) reflects this simplicity. However, the security policy for a trusted system, as argued above, is much more comprehensive. The definition of security that a trusted system must enforce may not be well–understood by the user or the developer. The trusted system's FMSP, then, must provide this definition in a clear and concise exposition. Certain pieces of evidence, such as the FMSP, play a more critical role in the development of a trusted system than they do in a trusted product, because they help clarify the design and implementation goals for the trusted system's development.

5 The Handbook

The Handbook targets a diverse audience, including acquisition managers, program managers, evaluators and users. This effort is based not only on NRL's experience in computer security evaluation but on its experience and research in software engineering, formal methods and computer security.

The Handbook will examine each piece of certification evidence required by the TCSEC for the assurance that it can provide during the development of a trusted system. The primary reader of this handbook (a trusted system evaluator) should gain significant insights into the contribution of each piece of evidence to the assurance argument. NRL has chosen the B3 class for study because we want to investigate the different kinds of assurance provided by different development strategies, and this class includes rigorous design and engineering approaches as well as formal methods.

A B3 TCB is a painstakingly crafted, tightly engineered set of software and hardware. However, as discussed above, engineering the TCB constitutes only part of the effort — the assurance argument consumes a significant portion of the development resources. To illustrate the importance of this argument for high assurance systems, the primary difference between the B3 class and the A1 class is the presentation of the assurance argument. For example, the B3 class requires only an informal specification of the TCB interface and an informal argument that this interface satisfies the requirements of the FMSP. At the A1 class, both the specification and its corresponding argument must be expressed formally. As one advances from the lowest class through the higher classes of the TCSEC, the assurance argument becomes more important to the certification effort.

The Handbook consists of a chapter for each piece of certification evidence, as well as an overview chapter with a sample plan for certifying a B3 trusted system. There is also a separate chapter for the trusted system development plan. The certification evidence for a B3 TCB includes:

- existence and documentation of a configuration management system
- a Formal Model of the Security Policy (FMSP)
- a Descriptive Top Level Specification (DTLS) of the TCB's interface
- a detailed design
- a demonstration that the DTLS is consistent with the FMSP
- an implementation (source code and related documents)
- a demonstration that the implementation is consistent with the DTLS
- a covert channel analysis
- security features testing
- penetration testing
- a Security Features User's Guide
- a Trusted Facility Manual

Each chapter provides an overview for the program manager, a precise description of the evidence and its purposes, and a detailed tutorial for evaluating the evidence. There are several questions that each chapter must address. Those questions (as applied to the FMSP) are below:

- *What is an FMSP?* The composition and structure of the FMSP are described along with its primary purpose and use. This question is actually answered twice — a broad view of the FMSP is provided for the program manager as part of the planning guidance, and then a specific description is provided for the evaluator. For example, the latter discussion includes:
  - what is meant by "formal",
  - the role of the computational model and some examples,
  - the role of the definition of security and how it can be expressed in the computational models discussed previously,
  - the role of assumptions in an FMSP, and finally,
  - the FMSP's contribution to the assurance argument, including its correspondence to the trusted system security policy and how it facilitates the development of future evidence (e.g., the DTLS) through the construction of a concrete model.
- *What is its purpose, and what makes a good one?* At least three parties examine the FMSP during the development process: the developer, the user and the evaluator. Each party uses the FMSP for different purposes. Each purpose is identified, and the qualities of the FMSP that fulfill that purpose are discussed.
- *How does the FMSP fit into the development life cycle?* No piece of evidence is meant to be produced and put on a shelf. Its impact on the subsequent system development as well as other certification evidence must be explored. The FMSP, for example, states the formal definition of security for the trusted system, and this definition should be interpreted for each stage of the trusted system's design.

- *What information is used to develop the FMSP?* Similarly, the impact of earlier system development activities and certification evidence on the development of the FMSP is discussed.

- *How do you evaluate the FMSP?* The bulk of what the evaluator should understand about the evidence has been presented already. This discussion concentrates on tactics and strategies for discovering whether the evidence satisfies its requirements and fulfills its purposes.

- *What overall assurance does the FMSP provide? What assurance does the FMSP provide before the system is built?* These questions are addressed throughout the chapter.

Each chapter also includes a sample assessment based on the discussion therein.

The Handbook addresses some of the fundamental issues of assurance, so it is not an evaluation checklist. The evaluator should not expect to find a list of tasks to perform when assessing a piece of certification evidence, but he or she should expect to find the answers to questions that might arise during the evaluation. Each trusted system is unique, so each trusted system evaluation is unique as well. An evaluation checklist is insufficient for all circumstances, because the trusted system's operational environment determines, to a large degree, what assurance must be provided.

Writing a handbook that addresses all possible evaluation and security policy scenarios is impractical. Instead, NRL tries to identify the primary contribution of each piece of certification evidence to the trusted system's assurance argument. It is likely that the evaluator will have to continue researching elsewhere when evaluating some evidence, but we hope that the information in this handbook will suggest the questions that motivate that search.

6 Summary

An assurance argument is a critical part of a trusted system certification, particularly for those systems evaluated at the higher levels of the TCSEC. NRL hopes that its study of the TCSEC's certification evidence will yield a better understanding of assurance and in turn, a better approach for achieving it.

References

[1] National Computer Security Center, Ft. Meade, MD, *DoD 5200.28-STD, Trusted Computer System Evaluation Criteria*, December 1985.

[2] Commission of the European Communities, Luxembourg, *Information Technology Security Evaluation Criteria (ITSEC)*, June 1991.