Rainbows and Arrows : How the Security Criteria Address Computer Misuse

Peter G. Neumann
Computer Science Lab, SRI International, Menlo Park CA 94025-3493
Neumann@csl.sri.com Phone 415-859-2375

13th National Computer Security Conference
Washington DC, 1-4 October 1990

Abstract

This paper examines the two main sets of computer security evaluation criteria and considers the extent to which each criterion combats various types of threats. Differences among the criteria sets are summarized, and recommendations are offered for improved coverage.

Introduction

In the 1989 National Computer Security Conference, Neumann and Parker [89] considered various classes of techniques for intentional or accidental misuse of computers and communications. Table 1 gives a terse summary of the misuse classes and illustrations of various types of misuse techniques. Exploitations frequently involve multiple techniques used in combination.

Two sets of security criteria are considered here, the U.S. Trusted Computer Security Evaluation Criteria (TCSEC) and the European Information Technology Security Evaluation Criteria (ITSEC). Each has certain strengths and certain deficiencies. Together they remain incomplete in their coverage and not completely consistent with one another. Nevertheless, they must be considered as part of an evolutionary process, and represent important steps toward improved system security.

Both criteria sets are threat oriented. They are themselves evaluated here with respect to the specific threats that they do or do not address.

The TCSEC

The Trusted Computer Security Evaluation Criteria (TCSEC) of the United States Department of Defense are summarized in Figure 1, which is reproduced from TCSEC [85] (the Orange Book). Apart from the degenerate D class, each evaluation class (designated C1, C2, B1, B2, B3, A1, in order of generally increasing functionality and assurance) has associated with it a collection of criteria that address security policy, accountability, assurance, and documentation. The criteria for any evaluation class subsume the criteria at lesser classes. Many of the criteria elements have different implications at different evaluation classes; for example, security testing appears in Figure 1 with increasingly stringent requirements at each evaluation class from C1 to A1. Overall, the C class corresponds to conventional threats, the B class to more severe threats, and the A1 class provides greater assurance for B3 functionality.

A distinction is made between the ratings of products and the security of installed systems. Actual configurations of systems, particularly when networked, may result in vulnerabilities in spite of the evaluated ratings of individual component products. For example, passwords transmitted in the clear between networked systems may permit easy system compromise. Similarly, a flawed *sendmail* can undermine systems through dial-up lines, even without networking. Consequently, it is vital to consider each system complex (including networks, distributed system control, database management, and applications) as a single system. For this purpose, the Trusted Network Interpretation (TNI, Red Book, TCSEC-TNI [87]) and the Trusted Database Interpretation (TDI, TCSEC-TDI [89]) should be considered in addition to the Orange Book, along with others in the 'rainbow' series of documents -- which help to put TCSEC [85] in context. Analysis of a composite system may benefit from component evaluations; however, because the TCSEC were established before composite systems had become better understood, there are some basic shortcomings.

The Harmonised ITSEC

The Information Technology Security Evaluation Criteria (ITSEC), the Harmonised Criteria of France, Germany, the Netherlands, and the United Kingdom (ITSEC [90]) represent an effort to establish a comprehensive set of security requirements for widespread international use. ITSEC is generally intended as a superset of TCSEC, with ITSEC ratings mappable onto the TCSEC evaluation classes (see below). Historically, ITSEC represents a remarkably facile evolutionary grafting together of the evaluation classes of the German [light] Green Book ('das grüne Buch', GISA [89]) and the 'claims language' of the British [dark] Green Books (DTI [89]). (The predecessor criteria are considered here only in passing. Brunnstein and Fischer-Huebner [90] and Pfleeger [90] contrast these criteria with TCSEC.)

- **EX:** External abuse
    1. Visual spying: observation of keystrokes or screens
    2. Misrepresentation: deception of operators and users
    3. Physical scavenging: dumpster-diving for printout
- **HW:** Hardware abuse
    4. Logical Scavenging: examining discarded/stolen media
    5. Eavesdropping: electronic or other data interception
    6. Interference: electronic or other jamming
    7. Physical attack on or modification of equipment or power
    8. Physical removal of equipment and storage media
- **MQ:** Masquerading
    9. Impersonation (false identity external to computer systems)
    10. Piggybacking attacks (on communication lines, workstations)
    11. Playback and spoofing attacks
    12. Network weaving to mask physical whereabouts or routing
- **PP:** 'Pest' programs (setting up further abuses)
    13. Trojan-horse attacks (including letter bombs)
    14. Logic bombs (including time bombs), a form of Trojan horse
    15. Malevolent worm attacks, acquiring distributed resources
    16. Virus attacks, attaching to programs and replicating
- **BY:** Bypassing authentication/authority
    17. Trapdoor attacks (due to any of a variety of sources) --
        a. Improper identification and authentication
        b. Improper initialization or allocation
        c. Improper termination or deallocation
        d. Improper validation
        e. Naming flaws, confusions, and aliases
        f. Improper encapsulation: exposed implementation detail
        g. Asynchronous flaws: time-of-check to time-of-use anomalies
        h. Other logic errors
    18. Authorization attacks (e.g., password cracking, token hacking)
- **AM:** Active misuse of authority (writing, using, with apparent authorization) --
    19. Creation, modification, use (including false data entry)
    20. Incremental attacks (e.g., salami attacks)
    21. Denials of service (including saturation attacks)
- **PM:** Passive misuse of authority (reading, with apparent authorization) --
    22. Browsing randomly or searching for particular characteristics
    23. Inference and aggregation (especially in databases), traffic analysis
    24. Covert channel exploitation and other data leakage
- **IM:** 25. Misuse through inaction: willful neglect, errors of omission
- **IN:** 26. Use as an indirect aid for subsequent abuse: off-line preencryptive matching, factoring large numbers, autodialer scanning.

Table 1: Summary of Techniques for Computer Misuse

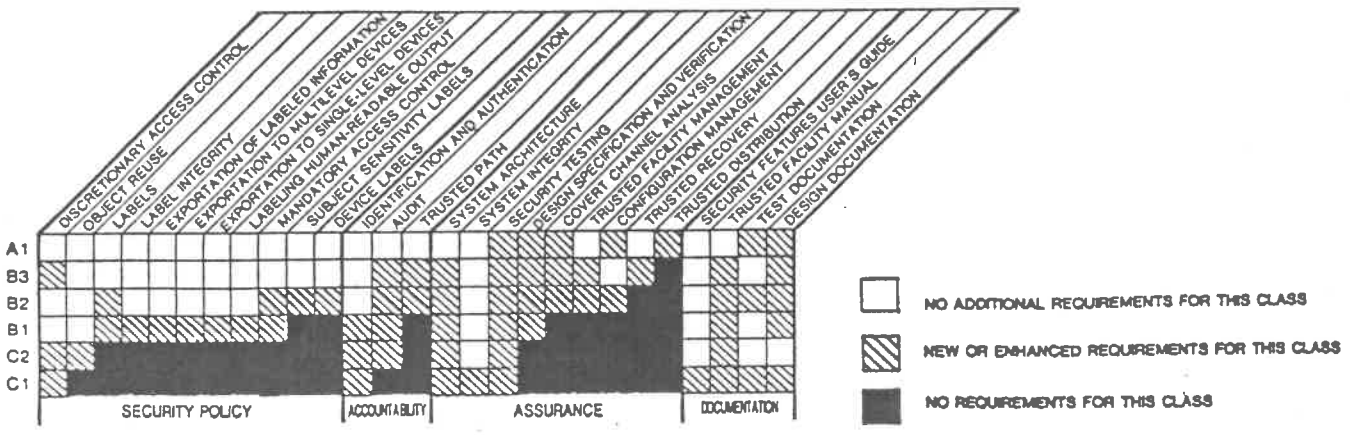

Figure 1: TCSEC Summary Chart

ITSEC unbundles *functional criteria* (F1 to F10) and *correctness criteria* (E0 as the degenerate case, and E1 to E6), which are evaluated independently.

The functional criteria F1 to F5 are of generally increasing merit, and correspond roughly to the functionality of C1, C2, B1, B2, and B3, respectively. The remaining functionality criteria address data and program integrity (F6), system availability (F7), data integrity in communication (F8), data confidentiality in communication (F9), and network security including confidentiality and integrity (F10). F6 to F10 may in principle be evaluated orthogonally to each other and to the chosen base level, F1, F2, F3, F4, or F5.

The correctness criteria are intended to provide increased assurance. To a first approximation, the correctness criteria cumulatively require testing (E1), configuration control and controlled distribution (E2), access to the detailed design and source code (E3), rigorous vulnerability analysis (E4), demonstrable correspondence between detailed design and source code (E5), and formal models, formal descriptions, and formal correspondences between them (E6). E2 through E6 correspond roughly to the assurance aspects of C2, B1, B2, B3, and A1, respectively.

An ITSEC rating is thus *one or none* of F1 to F5, *one* of E0 to E6, and *one or none of each* of F6 to F10, i.e., one of 6x7x2x2x2x2x2 = 1344 ratings. The intended approximate mappings from ITSEC functionality and correctness to TCSEC evaluation classes are given in Table 2 (although the respective definitions are not always completely consistent). F6 to F10 do not enter into the mapping, as they have no direct correspondence in TCSEC.

ITSEC function level	ITSEC correctness level	TCSEC evaluation class
	E0	D
F1	E2	C1
F2	E2	C2
F3	E3	B1
F4	E4	B2
F5	E5	B3
F5	E6	A1

Table 2: Mapping of ITSEC *onto* TCSEC

Because of the unbundling of functionality and assurance, other combinations such as F4/E3 are potentially meaningful. However, extreme combinations such as F5+6+7+8+9/E0 and F1/E6 are unrealistic. In any event, the mapping from ITSEC to TCSEC is many-to-one (e.g., F4/E3 and F3+7/E3 both map to B1), and therefore not uniquely reversible in the absence of the original ITSEC context (i.e., B1 maps back to F3/E3).

ITSEC's unbundling has advantages and disadvantages. On the whole it is a meritorious concept, as long as assurance does not become a victim of commercial expediency, and if the plethora of rating combinations does not cause confusion.

Although ITSEC [90] contains nothing analogous to Figure 1, there is a comparable table in the precursor German criteria document (GISA [89], pp. 106-107) for its functional and 'quality' (now 'correctness') criteria, distinguishing as in Figure 1 between 'no requirement', 'new requirement', and 'no new requirement'. Because Figure 1 is so useful as a definitional reference, a similar table would be useful for ITSEC.

ITSEC addresses "generic headings" of identification and authentication, access control, accountability, audit, object reuse, accuracy, reliability of service, and data exchange. A semiformal *claims language* is used to define particular properties that must be satisfied. The claims provide the basis for evaluation or self-evaluation. Table B.1 of ITSEC [90] shows the relationships between 35 claims-language "target phrases" and the generic headings.

The Criteria and the Misuse Techniques

This section contrasts TCSEC and ITSEC, and also discusses their applicability to the various misuse techniques. Table 3 indicates which misuse techniques (Table 1) are addressed by each of the two sets of evaluation criteria, TCSEC and ITSEC. The technique-type numbers and symbolic class designators in Table 3 are those noted in Table 1. An entry in the body of Table 3 implies that the particular criteria element contributes something constructive to the prevention or detection of the indicated misuse technique or class. However, because of the inherently weak-link nature of security, it is necessary to consider the coverage provided by the totality of all criteria rather than that of any individual criterion.

The first section of Table 3 summarizes the misuse techniques relative to the TCSEC evaluation criteria (Figure 1). Apart from questions of the extent of protection and assurance, the TCSEC entries of Table 3 are relatively independent of the specific evaluation classes, for those evaluation classes for which requirements exist (i.e., for which the matrix entry in Figure 1 is not black).

To the extent that the ITSEC criteria F1 to F5 map onto the TCSEC criteria (when combined with the correctness criteria, as noted in Table 2), the ITSEC functionality classes F1 to F5 can be related directly to the first section of Table 3 via the particular combination of TCSEC requirements in Figure 1. The additional functionality criteria (F6 to F10) are only partially covered by TCSEC and TNI. The relevance of ITSEC to the misuse techniques is summarized in the second section of Table 3.

Criterion \ Misuse Techniques	External	Hardware	Masquerading	Pests, Bypasses	Active Misuse	Browsing	Inference	Covert channels	Inaction	Indirect aid
	EX 1-3	HW 4-8	MQ 9-12	PP,BY 13-18	AM 19-21	PM 22	PM 23	PM 24	IM 25	IN 26
TCSEC Security Policy:										
Discretionary access control				(*)	*	*				
Object reuse				17b,c						
Labels & label integrity				*	*	*	*	*		
Exportation (3 criteria)		(4)		*	*	*	*	*		
Labeling human-read output	(3)			*	*	*	*	*		
Mandatory access controls				*	*	*	*	*		
Subject sensitivity labels				*	*	*	*	*		
Device labels				*	*	*	*	*		
TCSEC Accountability:										
Identification/authentication			(9)	*	*	*	(*)	(*)		
Audit			(*)	*	*	*	*	*	(*)	(*)
Trusted path			*	*						
TCSEC Assurance:										
System architecture			(*)	*	*	*	*	*		
System integrity				*						
Security testing			(*)	(*)						
Design spec/verification				*	*	*	(*)	*		
Covert channel analysis						(*)	(*)	*		
Trusted facility management	(3)			(*)	*	*	(*)			
Configuration management			(*)	*						
Trusted recovery			(*)	*						
Trusted distribution				*						
TCSEC Documentation:										
Security features user's guide				*	*	*	*	*		
Trusted facility manual				*	*	*	*	*		
Test documentation				*	*	*	*	*		
Design documentation				*	*	*	*	*		
ITSEC Functionality and Correctness:										
F1. C1 functionality				(*)	*	(*)				
F2. C2 functionality				(*)	*	(*)				
F3. B1 functionality	(3)	(4)		(*)	*	(*)				
F4. B2 functionality	(3)	(4)	(*)	*	*	*	*	*		
F5. B3 functionality	(3)	(4)	(*)	*	*	*	*	*	(*)	(*)
F6. Data/program integrity				*	*					
F7. System availability		(6-8)		*	*	(*)				
F8. Comm data integrity		6		(*)	*					
F9. Comm data confidentiality		5		(*)	*	*	*	(*)		
F10.Network security/integrity		5	*	*	*		(*)	(*)		
E1-E3, with varying assurance				(*)	(*)	(*)	(*)			
E4-E6, with varying assurance				(*)	(*)	(*)	(*)	*		

Legend: The column-head misuse class designators and the misuse-type numbers refer to those in Table 1.
Misuse types are grouped according to similar characteristics.
'*' implies the given criterion helps generally to combat the misuse class(es) in the column head.
Numbers imply only certain misuse types are applicable within the column-head class(es).
Parentheses imply a secondary effect for the particular criterion and misuse class(es) or type.
Refer to Figure 1 for relevant TCSEC evaluation classes for each TCSEC criterion.

Table 3: Criteria relevance for combatting misuse techniques

To the extent that ITSEC is a proper superset of TCSEC, many of the following comments about TCSEC are also relevant to ITSEC. When ITSEC is discussed *per se*, it is usually where it differs from TCSEC.

TCSEC bundles its criteria in two dimensions, as can be seen in Figure 1. First, functionality and assurance criteria are coupled rather rigidly. Second, each evaluation class is considered as a monolithic collection of criteria; in practice it would be useful to define intermediate evaluation classes such as 'C2+' or 'B1+", defined with certain specified features of higher classes and extra requirements (e.g., akin to F6 to F10).

The TCSEC criteria do not adequately address availability, data integrity (such as assurances that files have not been tampered with through bypasses to the write-protection mechanism), and generalized nondenial of service, for example. (The ITSEC criteria are somewhat more explicit about requiring availability and preventing denials of service.) The TCSEC criteria also do not address trusted paths to and authentication by virtual systems that do not have comparable facilities with respect to the end users, although extensions have been proposed. Furthermore, there is still some uncertainty about the criteria-relevant effects of layered trusted computing bases (TCBs). These considerations, together with the proliferation of TCSEC 'interpretations' (e.g., TNI and TDI for networks and databases, respectively), indicate that there are additions to TCSEC that would be relevant; indeed, the ITSEC F6-F10 have attempted to address some of them. All of the misuse techniques of Table 1 are relevant to distributed systems, networks of computer systems, and database systems, and thus need to be covered explicitly by any subsequent extensions or modifications to the criteria.

The ITSEC F1-F5 functional criteria (together with the appropriate correctness criteria) map fairly well onto the TCSEC requirements, according to Table 2, while F6-F10 do not. For F6 to F10, it is unclear what correctness criteria would be meaningful in isolation, particularly because failure to enforce the F6-F10 requirements with adequate assurance could actually undermine the enforcement of overall system security supposedly covered by the F1 to F5 rating. For example, inadequate attention to integrity, communications, or networking can undermine the security of installed computer systems. The *sendmail debug option* problem provides an illustration.

Defensive measures should be chosen to prevent wasteful coverage of nonthreats and to prevent gaps from existing at the interfaces among the various measures. Indeed, the 'Chinese Menu' flavor of the ITSEC criteria (i.e., the unbundling of functionality and correctness, plus the ostensibly orthogonal F6 to F10 requirements) appears to be attractive for that reason. However, many of 1344 potential ratings of ITSEC functionality and correctness are not particularly logical, consistent, or sound, and should be avoided; in contrast, the mapping (Table 2) of ITSEC ratings onto one of only seven TCSEC evaluation classes suggests that TCSEC might be too monolithic.

Further Discussion

Security requires an overall systems view, and all potential weak links must be considered. TCSEC and ITSEC focus on certain basic aspects of system misuse, but are less comprehensive in others. In this section we consider the roles of security policy, accountability, and assurance, as well as the special problems of networks and databases.

Security policy

Table 3 suggests that security policy criteria help to address the basic misuse types (13-24), but the table does not indicate the extent to which weak-link phenomena predominate. In particular, examination of the column for pest programs and bypasses shows that the problems of preventing these forms of attack are rather pervasive, in that *every one* of the criteria elements contributes something to combatting these attacks, but that in combination all of the criteria are still not quite enough. Penetrators typically appear as if they are authorized users. Pest programs are especially insidious because they execute on behalf of authorized users, with the normal privileges of their unsuspecting victims. Although any particular *known* personal computer virus (or propagating Trojan horse) that does not mutate may be detectable, viruses are *in general* very difficult to detect -- especially if they resort to techniques such as mutation, length-preserving compressions, and dispersion into small pieces. Such techniques escalate pest-program defense to being 'beyond feasibility' in general.

Thus, a combination of all of the cited criteria elements (including better PC hardware and operating systems) evidently would help somewhat, but would still not be enough. Finer-grain access controls that closely reduce what is permitted to just what is actually necessary can help to combat these attacks, including misuse by apparently authorized users, by narrowing the basic gap that otherwise prevents access controls from enforcing what is actually intended.

The existence of compartmented multilevel security (MLS) tends to limit some of the adverse effects from pest programs and bypasses -- notably adverse flow of information -- as well as reducing opportunities for misuse by authorized users. MLS is of potential value throughout a distributed system or network, assuming that there is comparable trustworthiness in enforcement. Some sort of mandatory integrity (e.g., the restrictive multilevel integrity, MLI, of Biba [75], or the more flexible type-based integrity provided by LOCK, Boebert [85]) can also help, particularly in preventing trusted applications from depending on less trusted programs and data, assuming

explicit or implicit certification of new programs and data. Denials of service could be restricted by the combination of MLS and MLI, at least by confining the effects within security/integrity levels and compartments. However, even with such multilevel controls there are still vulnerabilities, such as malicious deletion within the same level and compartment. The application integrity policy of Clark and Wilson [87] also provides a significant set of criteria, relating to good software engineering practice.

The scope of coverage is quite diverse for the various security-policy related criteria elements. One of the more narrowly defined requirements is the TCSEC criterion for proper object reuse, addressing improper initialization or allocation (type 17b) and also relating to improper termination or deallocation (type 17c) in Table 3. Its proper enforcement depends on noncompromisability of other criteria. For the general technique class of bypassing authentication and authority (BY) in Table 3, preventing the subtypes of trapdoor attacks (type 17) requires intelligent software development; object reuse is just one specific example of this need. Thus, implicit in the process of adhering to the criteria is a requirement that demands better system engineering, including software, hardware, and the operating environment. Also, inherently weak security policies (e.g., C2 discretionary access) should not be relied on in critical applications.

Accountability

Passwords provide a fundamentally flawed authentication mechanism, although neither criteria set adequately reflects the seriousness of the problems. For example, there is a B1 requirement for authentication, but nothing higher except for the trusted path requirements at B2 and B3, which only slightly reduce the threats to password compromise. Something more stringent (such as encryption-based authenticators) is undoubtably desirable in sensitive environments, although even those mechanisms are vulnerable to certain forms of compromise.

Logging and auditing play a vital role throughout. (Auditing is the only criterion that addresses the rather obscure techniques of misuse through inaction (type 25) and use as an indirect aid (type 26), and then only *after the fact*.) Although not addressed in detail in either of the criteria sets, real-time audit-trail analysis is expected to become a major contributor in the future, in hopes of catching perpetrators *in flagrante delicto*. Lunt [88] surveys real-time analysis systems that use rule-based expert systems and/or profile-based statistical systems. Real-time analysis has the potential of providing additional deterrents that *post-hoc* analysis cannot.

Nonrepudiation is a rather specific requirement (e.g., DTI [89]), addressing a small corner of the authentication problem in which authenticity cannot easily be denied at a later time, i.e., part of the attack technique of improper identification and authentication (type 17a). Nonrepudiation was present in the predecessor Dark Green books, but appears only implicitly in ITSEC.

Assurance

Examination of Table 3 indicates that the criteria only incidentally address external abuse and hardware abuse (technique types 1 to 8). Protection against emanations (part of type 5) and interference (type 6) is extensively covered elsewhere for military and intelligence applications, but is widely ignored in other applications. Nevertheless, stray emanations and interference have been responsible for human deaths in life-critical applications (e.g., the combination of microwave emanations and heart-pacemaker interference), and must be recognized as both security and integrity problems in critical environments. More generally, better administrative guidance for external and hardware abuse would be appropriate, particularly for unclassified critical applications.

Physical security is an important part of defending against various classes of attack; it is generally thought of in relation to hardware abuse and certain external attacks, but often is relied upon implicitly for authentication, trusted path, and configuration management criteria as well. It is usually considered separate from computer security, less glamorous in its nonresearch nature. However, physical access to computer and communication equipment can seriously undermine the ability to enforce TCB security and integrity. For example, the trusted distribution requirement (relating to some assurances that the system is untampered with) arises in TCSEC only at A1, but is generally relevant; trusted paths arise at B2 and B3, but are also meaningful below that. Defending against the insertion of pest programs and trap doors depends not only on the cited criteria but also to some extent on physical security and people, especially in personal computers.

Operational security is another serious concern. Trusted facility management has requirements at the B2 and B3 classes relating to the separation of duties between operator and administrator roles (B2) and additionally between security administrator and system administrator roles (B3). (Separation of duties more generally is fundamental to the Clark and Wilson integrity model, and is not explicitly addressed by either of the criteria sets.)

Databases

For database management systems, all of the basic misuse techniques and all of the criteria elements of Table 3 are relevant. Of particular importance are database issues relating to integrity, inference, and covert channels, at a granularity different from operating systems:

- Integrity constraints often lead to confusion in databases. Consistency of distributed and/or replicated data has both security and integrity implications. Primary-key uniqueness and referential integrity have both integrity and

inference implications. Integrity locks (e.g., cryptological seals) are of interest, although compromises via the underlying operating system must be considered.

• Inference issues are intrinsic in databases, and generally impossible to combat completely.

• Covert channels arise for a variety of reasons, including the use of shared indices, concurrency controls, resource exhaustion, recovery, shared devices, and naming conflicts. They are also a common side-effect of discretionary access control mechanisms. They are seemingly more difficult to control in databases than in operating systems, especially when data dependent.

The TCSEC Trusted Database Interpretation (TCSEC-TDI [89]) gives considerable guidance on such issues. Databases are of interest to the ITSEC criteria only as instances of entire systems. Issues of hierarchically layered assurance raised by the TDI are in the long run likely to be very important, whenever systems are composed out of components with different degrees of trustworthiness. The absence of explicit layering of TCBs in the ITSEC criteria suggests that the entire DBMS and underlying operating system might have to be evaluated as one, rather than being able to reason about the underlying TCB. Nevertheless, compositional reasoning is plausible within ITSEC. (Discussion of balanced assurance versus uniform assurance must await the final version of the TDI.)

As an example of a database system targeted for a TCSEC B3 or A1 rating, the SeaView security/integrity model (Denning et al. [88]) and system design (Lunt et al. [88]) provide a general approach capable of advanced database security, including multilevel security. The SeaView architecture involves layers of trustworthiness, based on a multilevel secure trusted computing base (Gemini's GEMSOS), and a slightly modified commercial DBMS (Oracle). The database engine is untrusted for multilevel security, but is trusted for integrity. SeaView explicitly addresses a wide range of security and integrity threats (including misuse techniques 13 through 24).

Networks

The TNI and ITSEC F8, F9, and F10 are particularly relevant to networks; essentially all of the misuse techniques are applicable to computer-communication networks *per se* (irrespective of the computer systems that they conjoin), although the coverage in Table 3 is somewhat spotty. For example, the MILNET terminal access concentrators (TACs) provide dial-up or hard-wired access to all systems on MILNET. The TACs and the interface message processors (IMPs) are logically internal to the network, and invisible to ordinary programmers. Because the TACs and IMPs are systems (nodes) without 'users', some of the techniques may at first seem less applicable, such as the ability of unauthorized people to insert pest programs. However, such vulnerabilities still exist, because of the way in which program maintenance is done remotely using the network itself. This suggests that without careful consideration it is dangerous to assume that any of the criteria elements is *not* applicable. The 27 Oct 1980 Arpanet collapse and the 15 Jan 1990 AT&T slowdown both indicate how a system flaw can accidentally result in the propagation of damage throughout the network. (See Neumann [90a].) There is also an important security lesson to be learned from such accidental problems, because both could alternatively have been triggered intentionally. Thus, networking appears to require still broader coverage of vulnerabilities.

Other Criteria

The Canadian draft criteria (CSE [89]) outline classes A, B, C, and D, as in TCSEC, as well as divisions of integrity (E,F,G,H), availability (J,K,L,M), accountability (P,Q,R,S), and trustworthiness (T0, T1, T2, etc.). Draft French criteria also exist, in the ''Blue-White-Red Book''. (Harmonization of those two criteria sets could result in colorful gourmet alphabet soup. Vive la différence!)

The British Ministry of Defence has established a different set of standards (MoD [89]) for safety-critical computer systems. Those criteria require significant use of 'semiformal' methods. Indeed many of the requirements for secure systems are also relevant for life-critical systems, but by no means sufficient.

Conclusions

Table 3 represents an oversimplified effort to capture the essence of the relationships between the two criteria sets and the threats they seek to address. The reality is obviously more multidimensional, with some subtle distinctions among the different evaluation classes and among the different technique types within each misuse class. Nevertheless, the intent of this paper is to educe the major issues for deeper examination.

The two criteria sets have tended to focus to date largely on the simplest threats in relatively homogeneous systems. However, as technology and assurance measures both improve, as distributed sytems become more widespread, and as sophistication on the part of misusers increases, the serious threats may tend to change in nature and escalate in technology. Thus, it is important to anticipate such trends and ensure that the criteria cover all of the realistic threats. In general, this may result in a slow migration to intermediate or even higher functionality and assurance (whether currently defined or not), even in personal computers and workstations, and with particular attention to distributed systems and networking.

TCSEC and ITSEC are seen here to play useful roles in

combatting malicious (and to some extent unintentional) misuse of computer systems and networks. However, both criteria sets reflect some vestiges of their historical perspective (despite the recency of ITSEC); important classes of misuse and various advanced architectures are not adequately covered. In addition, there are many related issues that are shortshrifted, such as more explicit recognition of the software-engineering relevance of the application integrity policy of Clark and Wilson, the importance of reusability and composability of sound building blocks such as TCSEC TCBs, and the fundamental nature of authentication. Trustworthy identification and authentication are vital to distributed systems. Also important are generality and flexibility in evaluation of real systems, e.g., evaluating system products and modifications generically, while also evaluating specific installations in their live environments. Neither of the two criteria sets deals satisfactorily with the assurance that results from hooking together either homogeneous or heterogeneous system components, reflecting the vulnerabilities in layered, networked, and distributed systems, although the Trusted Network Interpretation (TNI) and ITSEC F8-F10 attempt to address these issues. Some major remaining research issues were exposed in the TDI attempts to properly address layering of trusted components, and must be resolved.

Recommended Extensions

Both sets of criteria represent significant efforts to improve security in general, and to reduce the risks of malicious misuse in particular. Several specific recommendations for desirable extensions are noted below.

- Further system integrity is desirable above C1, e.g., to hinder system tampering.
- Mandatory integrity mechanisms can reduce the dependence on untrustworthy code and data.
- Application integrity *a la* Clark-Wilson can limit malicious code and other problems in applications.
- Availability measures (including the use of integrity requirements) can limit denials of service by both unauthorized and authorized users.
- Higher-assurance authentication is desirable above B1. Passwords generally have too many vulnerabilities.
- Trusted distribution is desirable below A1; trusted recovery is desirable below B3; trusted paths may be relevant below B2. All three of these can have significant roles in the prevention of pest programs, even in C2 systems.
- Real-time audit-trail analysis has the potential to detect pest program hatching, penetrations, and misuses of authority preliminary to or concurrent with misuse.
- Greater attention to distinctions between products and operational systems is desirable, including more administrative and management guidelines, as well as the ability to accommodate compositions of evaluated products, installed systems, and incremental changes. Further guidance on how the criteria might help *installed systems* to prevent misuse would be very valuable.
- Composite systems must be addressed systematically.
- More attention should be given to formal code verification and other means of demonstrating whether code is consistent with its specifications. This is important in certain critical applications, not just for security but also where human safety and very high availability are vital. It is interesting that the precursor German criteria (GISA [89]) included a quality criterion Q7 (equivalent in spirit to the first incarnation of TCSEC's 'beyond A1'), which failed to harmonize into an ITSEC E7 ('beyond E6').
- Specific products and installed systems need to be better matched with the threats they are intended to address, addressing risks and cost benefits. Table 3 must be recognized as a superficial first step.
- ITSEC provides many challenges for an evolutionary next-generation TCSEC addressing the above points, and especially the issue of TCSEC/ITSEC reciprocity. However, it will be important not to introduce circular dependencies or inconsistencies between the two.

The original version of this paper contained the following comment, in light of the rainbow-colored criteria books: "Although the number of still unused colors is rapidly dwindling, it is hoped that neither the intersection of the requirements nor the union of the colors red and green will be used, for that would result in a *little black book* for security." It appears that the considerable harmonization already achieved in the past year by ITSEC has significantly reduced that concern.

Various "CLEFs" (CESG-Licensed Evaluation Facilities) are being formed to carry out ITSEC evaluations. Recognizing the considerable advantages that can result from relatively unrestricted reciprocity, it is hoped that harmonization of U.S., U.K., and German interests (among others) can lead to accord and creative counterpoint among the at-least-treble CLEFs, particularly in staving off nationalistic self-interest.

Malicious misuse of computer systems can never be prevented completely, particularly when perpetrated by authorized users. Nevertheless, there are considerable benefits that can be gained from evaluations with respect to the criteria addressed above -- with the recognition that some threats are not covered in adequate detail. (Note that too much specificity is also a bad idea if it stifles design diversity and exacerbates different vendors' compatibility concerns.) Further work is urgently needed to refine and extend TCSEC and ITSEC into a unified, coherent, international, mutually useful, and modern set of criteria that more precisely address the vulnerabilities and threats to be avoided, including those in heterogeneous distributed systems. Rapid convergence on such a universal set of

criteria will be essential to the development of appropriate future products, systems, and their evaluations. TCSEC and ITSEC must not be considered rigidly as gospel (or as competitors), and must rapidly evolve together into a unified whole. It will be important in the future to incorporate security, integrity, availability, guaranteed performance, safety (cf. MoD [89]), and other vital requirements within a common composite-system framework (Neumann [90b]), and to be able to enforce whichever of those requirements are necessary, so that there will be greater assurance that critical systems can simultaneously satisfy the combined set of requirements. (For example, see Neumann [86]). However, we must never assume perfection on the part of the computer systems and their user communities, and must design and use the technology accordingly.

Acknowledgements

The author is indebted to Teresa Lunt and Marjory Blumenthal for their helpful suggestions. This paper was prepared with support from National Science Foundation Grant CCR-8715419.

References

K.J. Biba [75], Integrity Considerations for Secure Computer Systems. Report MTR 3153, MITRE Corp., Bedford, Massachusetts, June 1975.

E. Boebert [85], A Practical Alternative to Hierarchical Integrity Policies. *Proc. Eighth National Computer Security Conference*, 30 September 1985.

K. Brunnstein and S. Fischer-Huebner [90], Risk Analysis of "Trusted Computer Systems", *Proc. 6th International Conference on Information Security: SEC'90*, IFIP TC-11, Helsinki-Espoo, 23-25 May 1990.

D. Clark and D. Wilson [87], A Comparison of Commercial and Military Computer Security Policies. *Proc. 1987 IEEE Symposium on Security and Privacy*, Oakland, California, April 1987, pp. 184-194.

CSE [89], Communications Security Establishment, Canadian Trusted Computer Product Evaluation Criteria. Draft, version 1.0, May 1989.

DTI [89], Commercial Computer Security Centre, Department of Trade and Industry, volumes V01 (Overview Manual), V02 (Glossary), V03 (Index), V11 (Users' Code of Practice), V21 (Security Functionality Manual), V22 (Evaluation Levels Manual), V23 (Evaluation and Certification Manual), V31 (Vendors' Code of Practice), Version 3.0, February 1989.

D.E. Denning, T.F. Lunt, R.R. Schell, W.R. Shockley, M. Heckman [88], The SeaView Security Model. *Proc. 1988 IEEE Symposium on Security and Privacy*, April 1988, pp. 218-233.

GISA [89], IT-Security Criteria, Criteria for the Evaluation of Trustworthiness of Information Technology (IT) Systems. German Information Security Agency (ZSI), 11 January 1990.

ITSEC [90], Information Technology Security Evaluation Criteria, Harmonised Criteria of France, Germany, the Netherlands, and the United Kingdom. Draft Version 1, 2 May 1990. Available from UK CLEF, CESG Room 2/0805, Fiddlers Green Lane, Cheltenham U.K. GLOS GL52 5AJ, or ZSI, Am Nippenkreuz 19, D 5300 Bonn 2, West Germany.

T.F. Lunt [88], Automated Audit Trail Analysis and Intrusion Detection: A Survey. *11th National Computer Security Conference*, Baltimore, Maryland, 1988.

T.F. Lunt, R.R. Schell, W.R. Shockley, M. Heckman, D. Warren [88], A Near-Term Design for the SeaView Multilevel Database System. *Proc. 1988 IEEE Symposium on Security and Privacy*, April 1988, pp. 234-244.

MoD [89], Requirements for the procurement of safety critical software in defence equipment. Interim Defence Standard 00-55, Ministry of Defence, Directorate of Standardization, Kentigern House, 65 Brown St., Glasgow G2 8EX Scotland, U.K., May 1989.

P.G. Neumann [86], On Hierarchical Design of Computer Systems for Critical Applications. *IEEE Trans. Software Engineering* SE-12 9, September 1986, pp. 905-920.

P.G. Neumann [90a], The Computer-Related Risk of the Year: Distributed Control. *Proc. 5th COMPASS (IEEE)*, June 1990.

P.G. Neumann [90b], Towards Standards and Criteria for Critical Computer Systems. *Proc. 5th COMPASS (IEEE)*, June 1990.

P.G. Neumann and D.B. Parker [89], A Summary of Computer Misuse Techniques. *Proceedings of the 12th National Computer Security Conference*, Baltimore MD, 10-13 October 1989, pp. 396-407.

C.P. Pfleeger [90], Comparison of Trusted Systems Evaluation Criteria of the U.S., Germany, and Britain, *Proc. 5th COMPASS (IEEE)*, June 1990; based on earlier TIS Report #309, Trusted Information Systems, Inc., PO Box 45, Glenwood MD 21738, 2 March 1990.

TCSEC [85], Department of Defense Trusted Computer System Evaluation Criteria. DOD 5200.28-STD, December 1985 (Orange Book).

TCSEC-TDI [89], Trusted Database Management System Interpretation of the Trusted Computer System Evaluation Criteria. Draft, 25 October 1989, National Computer Security Center. (Revision pending.)

TCSEC-TNI [87], Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria. NCSC-TG-005 Version-1, 31 July 1987 (Red Book). (Revision pending.)

CIVIL AND MILITARY APPLICATION OF TRUSTED SYSTEMS CRITERIA

William C. Barker
Charles P. Pfleeger

Trusted Information Systems, Inc.
3060 Washington Road (Route 97)
Glenwood, MD 21738

Abstract

Trusted computer systems are commonly advertised in the context of military security requirements. Systems for military applications are designed to control access by need to know, by compartments, and by hierarchical levels. By introducing and defining a set of modes of operation for computer systems, then providing minimum evaluation criteria for systems operating in each mode, *Department of Defense (DoD)* guidelines can also be useful for civilian applications. To date, most development of trusted systems has been in response to military requirements, and the terminology and perceived usefulness of trusted systems development has tended to reflect this origin. It is, however, straightforward to map civilian needs—both functionality and assurance—onto military trusted systems.

Introduction

Many managers of non-military computer systems who perceive a need for computer security are frustrated by the scarcity of guidelines for selecting and applying trusted systems outside the defense and intelligence communities. The U.S. *DoD Trusted Computer Systems Evaluation Criteria [TCSEC]* is the only generally accepted criteria for evaluating the trustworthiness of computer systems in the United States. Trusted computer systems are commonly advertised in the context of military security requirements. Degrees of trustworthiness are expressed as digraphs (e.g., C2, B2, A1). Systems with lower ratings (e.g., C2) are designed to prevent the access to information by persons not having a legitimate "need to know" for that information. Systems granted higher ratings (i.e., B1) are designed to prevent the access to "compartmented" information by users not briefed into the "compartment," while even higher rated systems (i.e., B2, B3, and A1) are designed to prevent access to classified information by users not possessing security clearance for that information. However, there is little guidance concerning relevance of the *TCSEC* and attendant applications guidelines [CSC003] to civil requirements.

The DoD *Security Requirements for Automated Information Systems* [DoD28], while more overtly military in its audience, may provide a better foundation for determining trusted systems requirements than the National Computer Security Center's applications guidelines. By introducing and defining a set of modes of operation for computer systems, then providing minimum TCSEC ratings for systems operating in each mode, the DoD security requirements provide guidelines that are also useful for civilian applications. This paper is a brief description of different security attributes that may be associated with computer systems and the effects of those attributes on the modes in which the systems may safely operate. It attempts to treat both civil and defense applications classes, and to relate *TCSEC* features and assurances to both civilian and military requirements.

General Threats to Computer Systems

In the defense environment, disclosure is usually judged to be the most significant threat to computer systems. Public law requires the protection of certain kinds of information, especially that related to the national defense. For truly sensitive information, individuals are allowed the discretion to choose to whom they will release information only within the narrowly prescribed limits of security clearances. It is considered serious when information is disclosed to a person with an appropriate clearance but without the necessary need to know for the information. Because such recipients have been properly investigated and are trusted to protect similarly sensitive information, the disclosure is not considered extremely severe. However, if information is released to an unauthorized, untrusted person, the disclosure is deemed very serious because it may be tantamount to disclosing the information to a hostile agent. Through such a disclosure, physical assets may be lost, a competitive advantage may be compromised, or an expensive recovery may be occasioned.

In a medical example, sensitive information may be protected by being divided into several groups: physician, accounting, patient record, statistical, and so forth. Within each group, some individuals will have the right to see more information than others. The physician may have some notes that are strictly for her reference, others to be shared with colleagues in a physician