Paper Keywords and Abstracts/Introductions

This file lists some brief information about the papers very briefly. Unlike the files containing the actual papers, this file is searchable. Each entry includes:

- the name of the file on the CD-ROM containing the paper;
- bibliographic information, showing a formal citation;
- related papers adding to the topic and were either written as part of the same project or are the standard reference for the work;
- keywords; and
- abstract, preface, or introduction taken directly from the paper.

ande72.pdf

Bibliographic Information

James P. Anderson, *Computer Security Technology Planning Study Volume II*, ESD-TR-73-51, Vol. II, Electronic Systems Division, Air Force Systems Command, Hanscom Field, Bedford, MA 01730 (Oct. 1972).

Keywords

security kernel, reference monitor, Trojan horse, penetration, disclosure

Abstract or Introduction

Details of a planning study for USAF computer security requirements are presented. An Advanced development and Engineering program to obtain an open-use, multilevel secure computing capability is described. Plans are also presented for the related developments of communications security products and the interim solution to present secure computing problems. Finally a Exploratory development plan complementary to the recommended Advanced and Engineering development plans is also included.

ande80.pdf

Bibliographic Information

James P. Anderson, *Computer Security Threat Monitoring and Surveillance*, James P. Anderson Co, Fort Washington, PA (1980).

Keywords

audit, log, surveillance, monitoring, variation, intrusion detection

Abstract or Introduction

1.1 Introduction
This is the final report of a study, the purpose of which was to improve the computer security auditing and surveillance capability of the customer's systems.

1.2 Background
Audit trails are taken by the customer on a relatively long term (weekly or monthly) basis. This data is accumulated in conjunction with normal systems accounting programs. The audit data is derived from SMF records collected daily from all machines in the main and Special Center. The data is temporarily consolidated into a single file (“dump” data set) from which the various summary accounting and audit trail reports are produced. After the various reports are generated, the entire daily collection of data is transferred to tape. Several years of raw accounting data from all systems are kept in this medium.

Audit trail data is distributed to a variety of individuals for review: a DAC for GIMS applications, activity security officers for some applications located under their purview, but the majority to the customers’ data processing personnel! For the most part the users and sponsors of a data base or an application are not the recipients of security audit trail data.

Security audit trails can play an important role in the security program for a computer system. As they are presently structured, they are useful primarily in detecting unauthorized access to files. The currently collected customer audit trails are designed to detect unauthorized access to a dataset by user identifiers. However, it is evident that such audit trails are not complete. Users (particularly ADP “personnel" with direct programming access to datasets) may operate at a level of control that bypasses the application level auditing and access controls. In other systems, particularly data management systems, the normal mode of access is expected to be interactive. Programmers with the ability to use access method primitives can frequently access database files directly without leaving any trace in the application access control and audit logs. Under the circumstances, such audit trail concepts can do little more than attempt to detect frontal attacks on some system resource.

Security audit trails can play an important role in a security program for a computer system. As audit trails are presently structured on most machines, they are only useful primarily in detecting unauthorized access to files. For those computers which have no access control mechanisms built into the primary operating systems, the audit trail bears the burden of detecting unauthorized access to system resources. As access control mechanisms are installed in the operating systems, the need for security audit trail data will be even greater: it will not only be able to record attempted unauthorized access, but will be virtually the only method by which user actions which are authorized but excessive can be detected.

1.3 Summary

In computer installations in general, security audit trails, if taken, are rarely complete and almost never geared to the needs of the security officers whose responsibility it is to protect ADP assets. The balance of this report outlines the considerations and general design of a system which provides an initial set of tools to computer system security officers for use in their jobs. The discussion does not suggest the elimination of any existing security audit data collection and distribution. Rather it suggests augmenting any such schemes with information for the security personnel directly involved.

bell76.pdf

Bibliographic Information

David E. Bell and Leonard J. LaPadula, *Secure Computer System: Unified Exposition and MULTICS Interpretation*, MTR-2997 Rev. 1, The MITRE Corporation, Bedford, MA 01730 (Mar. 1976); also ESD-TR-75-306, rev. 1, Electronic Systems Division, Air Force Systems Command, Hanscom Field, Bedford, MA 01731.

Related Papers

- David E. Bell and Leonard J. LaPadula, "Secure Computer Systems: Mathematical Foundations," ESD-TR-73-278, Vol. I, Electronic Systems Division, Air Force Systems Command, Hanscom AFB, Bedford, MA (Nov. 1973).
- Leonard J. LaPadula and David E. Bell, "Secure Computer Systems: A Mathematical Model," ESD-TR-73-278, Vol. II, Electronic Systems Division, Air Force Systems Command, Hanscom AFB, Bedford, MA (Nov. 1973).
- David E. Bell, "Secure Computer Systems: A Refinement of the Mathematical Model," ESD-TR-73-278, Vol. III, Electronic Systems Division, Air Force Systems Command, Hanscom AFB, Bedford, MA (Apr. 1974).

Keywords

security policy, model simple security condition, star property, asterisk-property, mathematical model, secure computer system, security, trusted subject

Abstract or Introduction

For the past several years ESD has been involved in various projects relating to secure computer systems design and operation. One of the continuing efforts, started in 1972 at MITRE, has been secure computer system modeling. The effort initially produced a mathematical framework and a model [1, 2] and subsequently developed refinements and extensions to the model [3] which reflected a computer system architecture similar to that of Multics [4]. Recently a large effort has been proceeding to produce a design for a secure Multics based on the mathematical model given in [l, 2, 3].

Any attempt to use the model, whose documentation existed in three separate reports until this document was produced, would have been hampered by the lack of a single, consistent reference. Another problem for designers is the difficulty of relating the abstract entities of the model to the real entities of the Multics system. These two problems are solved by this document.

All significant material to date on the mathematical model has been collected in one place in the Appendix of this report. A number of minor changes have been incorporated, most of them notational or stylistic, in order to provide a uniform,

consistent, and easy-to-read reference. A substantive difference between the model of the Appendix and that of the references [2, 3] is the set of rules: the specific rules presented in Appendix have been adapted to the evolving Multics security kernel design.

Because the model is by nature abstract and, therefore, not understandable in one easy reading, Section II gives a prose description of the model.

In order to relate the mathematical model to the Multics design, Section III exhibits correspondences from Multics and security kernel entities to model entities.

Section IV discusses further considerations--topics which lie outside the scope of the current model but which are important issues for security kernel design.

As background for the remainder of this document, we briefly establish a general framework of related efforts in the rest of this section.

Work on secure computer systems, in one aspect or another, has been reported fairly continuously since the mid 1960s. Three periods are discernible: early history, transitional history, and current events.

The work by Weissmann [5] on the ADEPT-50 system stands out in the early history period. Not only was a fairly formal structuring of solution to a security problem provided, but ADEPT-50 was actually built and operated. In this early period the work of Lampson [6] is most representative of attempts to attack security problems rigorously through a formal medium of expression. In Lampson's work, the problem of access control is formulated very abstractly for the first time, using the concepts of "subjects," "object," and "access matrix." The early period, which ended in 1972, understandably did not provide a complete and demonstrable mathematical formulation of a solution.

The transitional period (1972 - 1974) is characterized by markedly increased interest in computer security issues as evidenced by the Anderson panel [7]. One of the principal results of this panel was the characterization of a solution to the problem of secure computing (using the concept of a "reference monitor") together with the reasoned dictum that comprehensive and rigorous modeling is intrinsic to a solution to the problem. This period also saw the development of the first demonstrated mathematical models [l, 2, 13] as well as ancillary mathematical results which characterized the nature of the correctness proof demonstration [2, 8]. A second modeling effort, also sponsored by the Electronic Systems Division of the United States Air Force and performed at Case-Western Reserve University, was also undertaken in this period [9]. In this model, the flow of information between repositories was investigated, initially in a static environment (that is, one in which neither creation nor deletion of agents or repositories is allowed) and subsequently in a dynamic environment. Many other papers appeared during this period. An implementation of a system based on a mathematical model was carried out at

MITRE by W. L. Schiller [10]. An extension and refinement of the first model was developed [3] to tailor the model to the exigencies of a proposed Multics implementation of the model; included in this extension was a concept promulgated at Case-Western Reserve concerning compatibility between the Multics directory structure and the classifications of the individual files. A great number of other computer security issues were investigated and characterized [11, 12, 13, 14, 15] during this time.

Current work succeeding the work reported above is a project sponsored by ESD and ARPA. In this project, the Air Force, the MITRE Corporation, and Honeywell are working cooperatively to develop a design for a security kernel for the Honeywell Multics (HIS level 68) computer system. Other significant efforts include work at UCLA [16], and the Stanford Research Institute [17].

This report summarizes, both narratively and formally, the particular version of the mathematical model that is relevant to the development of a Multics security kernel. The report not only presents the model in convenient and readable form, but also explicitly relates the model to the emerging Multics kernel design to help bridge the gap between the mathematical notions of the model and their counterparts in the Multics security kernel.

bisb78.pdf

Bibliographic Information

Richard Bisbey II and Dennis Hollingworth, *Protection Analysis: Final Report*, ISI/SR-78-13, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (May 1978).

Related Papers

- Richard Bisbey II, Gerald Popek, and James Carlstedt, "Protection Errors in Operating Systems: Inconsistency of a Single Data Value Over Time,"ISI/SR-75-4, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (Dec. 1975).
- Richard Bisbey II *et al*., "Data Dependency Analysis," ISI/RR-76-45, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (Feb. 1976).
- James Carlstedt *et al*., "Pattern Directed Protection Evaluation," ISI/RR-75-31, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (June 1975).
- James Carlstedt, "Protection Errors in Operating Systems: Validation of Critical Conditions," ISI/SR-76-5, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (May 1976).
- James Carlstedt, "Protection Errors in Operating Systems: A Selected Annotated Bibliography and Index to Terminology," ISI/SR-78-10, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (Jan. 1978).
- James Carlstedt, "Protection Errors in Operating Systems: Serialization," ISI/SR-78-9, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (Apr. 1978).
- Dennis Hollingworth and Richard Bisbey II, "Protection Errors in Operating Systems: Allocation/Deallocation Residuals," ISI/SR-76-7, University of Southern California/Information Sciences Institute, Marina Del Rey, CA 96291 (June 1976).
- Peter G. Neumann, "Computer Security Evaluation," 1978 National Computer Conference, AFIPS Conference Proceedings **47**, pp. 1087–1095 (1978).

Keywords

vulnerability, penetration, access control, error analysis, error-driven evaluation, error type, operating system security, protection evaluation, protection policy, software security

Abstract or Introduction

The Protection Analysis project was initiated at ISI by ARPA IPTO to further understand operating system security vulnerabilities and, where possible, identify automatable techniques for detecting such vulnerabilities in existing system software. The primary goal of the project was to make protection evaluation both

more effective and more economical by decomposing it into more manageable and methodical subtasks so as to drastically reduce the requirement for protection expertise and make it as independent as possible of the skills and motivation of the actual individuals involved. The project focused on near-term solutions to the problem of improving the security of existing and future operating systems in an attempt to have some impact on the security of the systems which would be in use over the next ten years.

A general strategy was identified, referred to as "pattern-directed protection evaluation" and tailored to the problem of evaluating existing systems. The approach provided a basis for categorizing protection errors according to their security-relevant properties; it was successfully applied for one such category to the MULTICS operating system, resulting in the detection of previously unknown security vulnerabilities.

dod85.pdf

Bibliographic Information

Department of Defense, *Trusted Computer System Evaluation Criteria*, DoD 5200.28-STD, National Computer Security Center, Ft. Meade, MD 20755 (Dec. 1985). Also known as the “Orange Book.”

Related Papers

- Department of Defense, *Password Management Guideline*, CSC-STD-002-85, National Computer Security Center, Ft. Meade, MD 20755 (Apr. 1985). Also known as the “Green Book.”
- Department of Defense, *Computer Security Requirements -- Guidance for Applying the DoD TCSEC in Specific Environments*, CSC-STD-003-85, National Computer Security Center, Ft. Meade, MD 20755 (June 1985). Also known as the “Light Yellow Book.”
- Department of Defense, *Technical Rational Behind CSC-STD-003-85: Computer Security Requirements -- Guidance for Applying the DoD TCSEC in Specific Environments*, CSC-STD-004-85, National Computer Security Center, Ft. Meade, MD 20755 (June 1985). Also known as the “Yellow Book.”
- Department of Defense, *A Guide to Understanding Audit in Trusted Systems*, Version 2, NCSC-TG-001 Ver. 2, National Computer Security Center, Ft. Meade, MD 20755 (June 1988). Also known as the “Tan Book.”
- Department of Defense, *Trusted Product Evaluations - A Guide for Vendors*, NCSC-TG-002, National Computer Security Center, Ft. Meade, MD 20755 (June 1990). Also known as the “Bright Blue Book.”
- Department of Defense, *A Guide to Understanding Discretionary Access Control in Trusted Systems*, NCSC-TG-003, National Computer Security Center, Ft. Meade, MD 20755 (Sep. 1987). Also known as the “Neon Orange Book.”
- Department of Defense, *Glossary of Computer Security Terms*, NCSC-TG-004, National Computer Security Center, Ft. Meade, MD 20755 (Oct. 1988). Also known as the “Teal Green Book.”
- Department of Defense, *Trusted Network Interpretation of the TCSEC (TNI)*, NCSC-TG-005, National Computer Security Center, Ft. Meade, MD 20755 (July 1987). Also known as the “Red Book.”
- Department of Defense, *A Guide to Understanding Configuration Management in Trusted Systems*, NCSC-TG-006, National Computer Security Center, Ft. Meade, MD 20755 (Mar. 1988). Also known as the “Amber Book.”
- Department of Defense, *A Guide to Understanding Design Documentation in Trusted Systems*, NCSC-TG-007, National Computer Security Center, Ft. Meade, MD 20755 (Oct. 1988). Also known as the “Burgundy Book.”
- Department of Defense, *A Guide to Understanding Trusted Distribution in Trusted Systems*, NCSC-TG-008, National Computer Security Center, Ft. Meade, MD 20755 (Dec. 1988). Also known as the “Dark Lavender Book.”

- Department of Defense, *Computer Security Subsystem Interpretation of the TCSEC*, NCSC-TG-009, National Computer Security Center, Ft. Meade, MD 20755 (Sep. 1988). Also known as the "Venice Blue Book."
- Department of Defense, *A Guide to Understanding Security Modeling in Trusted Systems*, NCSC-TG-010, National Computer Security Center, Ft. Meade, MD 20755 (Oct. 1992). Also known as the "Aqua Book."
- Department of Defense, *Trusted Network Interpretation Environments Guideline - Guidance for Applying the TNI*, NCSC-TG-011, National Computer Security Center, Ft. Meade, MD 20755 (Aug. 1990). Also known as the "Red Book."
- Department of Defense, *RAMP Program Document*, Version 2, NCSC-TG-013 Ver. 2, National Computer Security Center, Ft. Meade, MD 20755 (Mar. 1995). Also known as the "Pink Book."
- Department of Defense, *Guidelines for Formal Verification Systems*, NCSC-TG-014, National Computer Security Center, Ft. Meade, MD 20755 (Apr. 1989). Also known as the "Purple Book."
- Department of Defense, *A Guide to Understanding Trusted Facility Management*, NCSC-TG-015, National Computer Security Center, Ft. Meade, MD 20755 (Oct. 1989). Also known as the "Brown Book."
- Department of Defense, *Guidelines for Writing Trusted Facility Manuals*, NCSC-TG-016, National Computer Security Center, Ft. Meade, MD 20755 (Oct. 1989). Also known as the "Yellow-Green Book."
- Department of Defense, *A Guide to Understanding Identification and Authentication in Trusted Systems*, NCSC-TG-017, National Computer Security Center, Ft. Meade, MD 20755 (Sep. 1991). Also known as the "Light Blue Book."
- Department of Defense, *A Guide to Understanding Object Reuse in Trusted Systems*, NCSC-TG-018, National Computer Security Center, Ft. Meade, MD 20755 (July 1992). Also known as the "Light Blue Book."
- Department of Defense, *Trusted Product Evaluation Questionnaire*, Version 2, NCSC-TG-019 Ver. 2, National Computer Security Center, Ft. Meade, MD 20755 (May 1992). Also known as the "Blue Book."
- Department of Defense, *Trusted UNIX Working Group (TRUSIX) Rationale for Selecting Access Control List Features for the UNIX® System*, NCSC-TG-020-A, National Computer Security Center, Ft. Meade, MD 20755 (July 1989). Also known as the "Silver Book."
- Department of Defense, *Trusted Database Management System Interpretation of the TCSEC (TDI)*, NCSC-TG-021, National Computer Security Center, Ft. Meade, MD 20755 (Apr. 1991). Also known as the "Purple Book."
- Department of Defense, *A Guide to Understanding Trusted Recovery in Trusted Systems*, NCSC-TG-022, National Computer Security Center, Ft. Meade, MD 20755 (Dec. 1991). Also known as the "Yellow Book."
- Department of Defense, *A Guide to Understanding Security Testing and Test Documentation in Trusted Systems*, NCSC-TG-023, National Computer Security Center, Ft. Meade, MD 20755 (Dec. 1991). Also known as the "Bright Orange Book."

- Department of Defense, *A Guide to Procurement of Trusted Systems: An Introduction to Procurement Initiators on Computer Security Requirements*, Volume 1, NCSC-TG-024 Vol. 1, National Computer Security Center, Ft. Meade, MD 20755 (Dec. 1992). Also known as the "Purple Book."
- Department of Defense, *A Guide to Procurement of Trusted Systems: Language for RFP Specifications and Statements of Work - An Aid to Procurement Initiators*, Volume 2, NCSC-TG-024 Vol. 2, National Computer Security Center, Ft. Meade, MD 20755 (June 1993). Also known as the "Purple Book."
- Department of Defense, *A Guide to Procurement of Trusted Systems: Computer Security Contract Data Requirements List and Data Item Description Tutorial*, Volume 3, NCSC-TG-024 Vol. 3, National Computer Security Center, Ft. Meade, MD 20755 (Feb. 1994). Also known as the "Purple Book."
- Department of Defense, *A Guide to Understanding Data Remanence in Automated Information Systems*, Version 2, NCSC-TG-025 Ver. 2, National Computer Security Center, Ft. Meade, MD 20755 (Sep. 1991). Also known as the "Forest Green Book."
- Department of Defense, *A Guide to Writing the Security Features User's Guide for Trusted Systems*, NCSC-TG-026, National Computer Security Center, Ft. Meade, MD 20755 (Sep. 1991). Also known as the "Hot Peach Book."
- Department of Defense, *A Guide to Understanding Information System Security Officer Responsibilities for Automated Information Systems*, NCSC-TG-027, National Computer Security Center, Ft. Meade, MD 20755 (May 1992). Also known as the "Turquoise Book."
- Department of Defense, *Assessing Controlled Access Protection*, NCSC-TG-028, National Computer Security Center, Ft. Meade, MD 20755 (May 1992). Also known as the "Violet Book."
- Department of Defense, *Introduction to Certification and Accreditation Concepts*, NCSC-TG-029, National Computer Security Center, Ft. Meade, MD 20755 (Jan. 1994). Also known as the "Blue Book."
- Department of Defense, *A Guide to Understanding Covert Channel Analysis of Trusted Systems*, NCSC-TG-030, National Computer Security Center, Ft. Meade, MD 20755 (Nov. 1993). Also known as the "Light Pink Book."

Keywords

standard, trusted system, evaluation, Orange Book, protection, class, security requirement

Abstract or Introduction

The trusted computer system evaluation criteria defined in this document classify systems into four broad hierarchical divisions of enhanced security protection. They provide a basis for the evaluation of effectiveness of security controls built into automatic data processing system products. The criteria were developed with three objectives in mind: (a) to provide users with a yardstick with which to assess the degree of trust that can be placed in computer systems for the secure processing of classified or other sensitive information; (b) to provide guidance to manufacturers as to what to build into their new, widely-available trusted commercial products in

order to satisfy trust requirements for sensitive applications; and (c) to provide a basis for specifying security requirements in acquisition specifications. Two types of requirements are delineated for secure processing: (a) specific security feature requirements and (b) assurance requirements. Some of the latter requirements enable evaluation personnel to determine if the required features are present and functioning as intended. The scope of these criteria is to be applied to the set of components comprising a trusted system, and is not necessarily to be applied to each system component individually. Hence, some components of a system may be completely untrusted, while others may be individually evaluated to a lower or higher evaluation class than the trusted product considered as a whole system. In trusted products at the high end of the range, the strength of the reference monitor is such that most of the components can be completely untrusted. Though the criteria are intended to be application-independent, the specific security feature requirements may have to be interpreted when applying the criteria to specific systems with their own functional requirements, applications or special environments (e.g., communications processors, process control computers, and embedded systems in general). The underlying assurance requirements can be applied across the entire spectrum of ADP system or application processing environments without special interpretation.

ford78.pdf

Bibliographic Information

Ford Aerospace and Communications Corporation, *Secure Minicomputer Operating System (KSOS) Executive Summary: Phase I: Design of the Department of Defense Kernelized Secure Operating System*, WDL-781, Palo Alto, CA 94303 (Mar. 1978).

Keywords

trusted system, UNIX, formal specification, multilevel, security kernel, KSOS

Abstract or Introduction

The long-term goal of the KSOS effort is to develop a commercially viable computer operating system for the DEC PDP-11/70 that

- is compatible with the Bell Telephone Laboratories' UNIX*tm,
- is capable of efficiency comparable to standard UNIX*tm,
- enforces multilevel security and integrity, and
- is demonstrably secure.

In order to achieve this goal, the Phase I effort described here has designed a trusted Security Kernel and associated trusted Non-Kernel Security-Related Software, such that the trusted software:

- provides a suitable basis for KSOS;
- intrinsically supports multilevel security/integrity,
- can be used by itself to support non-UNIX*tm-based applications, and
- is able to run efficiently on a DEC PDP-11/70.

The security of the overall KSOS system must be convincingly demonstrated. This will be accomplished by formal verification of the security properties of the design (i.e., the formal specifications) and selected proofs of correspondence between the delivered code and the design. In addition, KSOS will be rigorously tested to lend added confidence in the in the system.

Although the Security Kernel is intended initially to support an Emulator providing a UNIX*tm-like user environment, the Kernel has been designed to be used by itself, or with an Emulator providing a different user environment. Typical uses of the Kernel by itself would be dedicated secure systems such as military message processing systems, or secure network front ends.

karg74.pdf

Bibliographic Information

Paul A. Karger and Roger R. Schell, *MULTICS Security Evaluation, Volume II: Vulnerability Analysis*, ESD-TR-74-193, Vol. II, Electronic Systems Division, Air Force Systems Command, Hanscom Field, Bedford, MA 01731 (June 1974).

Related Papers

- Ken Thompson, "Reflections on Trusting Trust," *Communications of the ACM* **27**(8) pp. 761-763 (Aug. 1984); Turing Award lecture.

Keywords

access control, multi-level system, operating system vulnerability, privacy, monitor, secure computer system, security kernel, penetration, security testing, segmentation

Abstract or Introduction

A security evaluation of Multics for potential use as a two-level (Secret/Top Secret) system in the Air Force Data Services Center (AFDSC) is presented. An overview is provided of the present implementation of the Multics Security controls. The report then details the results of a penetration exercise of Multics on the HIS 645 computer. In addition, preliminary results of a penetration exercise of Multics on the new HIS 6180 computer are presented. The report concludes that Multics as implemented today is not certifiably secure and cannot be used in an open use multi-level system- However, the Multics security design principles are significantly better than other contemporary systems. Thus, Multics as implemented today, can be used in a benign Secret/Top Secret environment . In addition, Multics forms a base from which a certifiably secure open use multi-level system can be developed.

lind76.pdf

Bibliographic Information

Theodore Linden, *Operating System Structures to Support Security and Reliable Software* NBS Technical Note 919, Institute for Computer Sciences and Technology, National Bureau of Standards, Department of Commerce, Washington DC 20234 (Aug. 1976).

Keywords

capability, capability-based addressing, extended-type objects, operating system structures, protection, reliable software, reliability, security, small protection domains, types.

Abstract or Introduction

Security has become an important and challenging goal in the design of computer systems. This survey focuses on two system structuring concepts that support security; namely, small protection domains and extended-type objects. These two concepts are especially promising because they also support reliable software by encouraging and enforcing highly modular software structures--in both systems software and in applications programs. Small protection domains allow each subunit or module of a program to be executed in a restricted environment that can prevent unanticipated or undesirable actions by that module. Extended-type objects provide a vehicle for data abstraction by allowing objects of new types to be manipulated in terms of operations that are natural for these objects. This provides a way to extend system protection features so that protection can be enforced in terms of applications-oriented operations on objects. This survey also explains one approach toward implementing these concepts thoroughly and efficiently--an approach based on the concept of capabilities incorporated into the addressing structure of the computer. Capability-based addressing is seen as a practical way to support future requirements for security and reliable software without sacrificing requirements for performance, flexibility, and sharing.

myer80.pdf

Bibliographic Information

Philip A. Myers, *Subversion: The Neglected Aspect of Computer Security*, Master Thesis, Naval Postgraduate School, Monterey CA 93940 (June 1980).

Keywords

subversion, protection policy, trap door, Trojan horse, penetration, access control, evaluation criteria, protection system, leakage of data, security kernel

Abstract or Introduction

This thesis distinguishes three methods of attacking internal protection mechanisms of computers: inadvertent disclosure, penetration, and subversion. Subversion is shown to be the most attractive to the serious attacker. Subversion is characterized by three phases of operations: the inserting of trap doors and Trojan horses, the exercising of them, and the retrieval of the resultant unauthorized information. Insertion occurs over the entire life cycle of the system from the system design phase to the production phase. This thesis clarifies the high risk of using computer systems, particularly so-called 'trusted' subsystems for the protection of sensitive information. This leads to a basis for countermeasures based on the lifetime protection of security related system components combined with the application of adequate technology as exemplified in the security kernel concept.

neum75.pdf

Bibliographic Information

James P. Anderson, *Computer Security Technology Planning Study Volume II*, ESD-TR-73-51, Vol. II, Electronic Systems Division, Air Force Systems Command, Hanscom Field, Bedford, MA 01730 (Oct. 1972).

Keywords

trusted system, formal specification, security kernel, PSOS, provably secure

Abstract or Introduction

This report summarizes work to date toward the development of a provably secure operating system. Discussed here are

- a methodology for the design, implementation, and proof of properties of large computing systems,
- the design of a secure operating system using this methodology,
- the security properties to be proven about this system,
- considerations for implementing such a system, and
- an approach to monitoring security and performance.

niba79.pdf

Bibliographic Information

Grace H. Nibaldi, *Proposed Technical Evaluation Criteria for Trusted Computer Systems*, M79-225, The MITRE Corporation, Bedford, MA 01730 (Oct. 1979).

Keywords

formal verification, classification, secure computer system, trusted computing base, evaluation criteria, evaluation process, policy, mechanism, assurance, level

Abstract or Introduction

The DoD has established a Computer Security Initiative to foster the widespread availability of trusted computer systems. An essential element of the Initiative is the identification of criteria and guidelines for evaluating the internal protection mechanisms of computer systems. This report documents a proposed set of technical evaluation criteria. These criteria and any evaluation process that they might imply represent one approach to how trusted systems might be evaluated.

scha75.pdf

Bibliographic Information

J. M. Schacht, *Jobstream Separator System Design*, MTR-3022 Vol. 1, The MITRE Corporation, Bedford, MA 01730 (May 1975).

Keywords

job stream separator, jobstream, isolation, security level, add on, reference monitor

Abstract or Introduction

The Jobstream Separator (JSS) has been proposed to automate the costly, inefficient, and inconvenient manual process utilized to "change colors" (security levels) at AF WWMCCS sites. The JSS would provide complete isolation among WWMCCS users and data at differing levels by introducing a secure, centralized, certifiably correct, minicomputer system to control electronic switching of peripheral devices during the system reconfiguration phase of the color change. The system would eliminate extensive operator intervention, reduce the delays incurred in the physical removal of storage media and enable the operator to change security states while maintaining overall security. This report presents a technical and economic assessment of the JSS and recommends development of a prototype system.

sche73.pdf

Bibliographic Information

Roger R. Schell, Peter J. Downey, and Gerald J. Popek, *Preliminary Notes on the Design of Secure Military Computer Systems*, MCI-73-1, The MITRE Corporation, Bedford, MA 01730 (Jan. 1973).

Keywords

secure computer system, secure model, secure design

Abstract or Introduction

The military has a heavy responsibility for protection of information in its shared computer systems. The military must insure the security of its computer systems before they are put into operational use. That is, the security must be “certified”, since once military information is lost it is irretrievable and there are no legal remedies for redress.

Most contemporary shared computer systems are not secure because security was not a mandatory requirement of the initial hardware and software design. The military has reasonably effective physical, communication, and personnel security, so that the nub of our computer security problem is the information access controls in the operating system and supporting hardware. We primarily need an effective means for enforcing very simple protection relationships, (e.g., user clearance level must be greater than or equal to the classification level of accessed information); however, we do not require solutions to some of the more complex protection problems such as mutually suspicious processes.

Based on the work of people like Butler Lampson we have espoused three design principles as a basis for adequate security controls:

a. Complete Mediation -- The system must provide complete mediation of information references, i.e., must interpose itself between any reference to sensitive data and accession of that data. All references must be validated by those portions of the system hardware and software responsible for security.

b. Isolation -- These valid operators, a “security kernel,” must be an isolated, tamper-proof component of the system. This kernel must provide a unique, protected identity for each user who generates references, and must protect the reference-validating algorithms.

c. Simplicity -- The security kernel must be simple enough for effective certification. The demonstrably complete logical design should be implemented as a small set of simple primitive operations and system database structures that can be shown to be correct.

These three principles are central to the understanding of the deficiencies of present systems and provide a basis for critical examination of protection mechanisms and a method for insuring a system is secure. It is our firm belief that by applying these principles we can have secure shared systems in the next few years.

schi75.pdf

Bibliographic Information

W. L. Schiller, *The Design and Specification of a Security Kernel for the PDP-11/45*, MTR-2934, The MITRE Corporation, Bedford, MA 01730 (Mar. 1975).

Keywords

security kernel, secure computer system, specification, model

Abstract or Introduction

This paper presents the design of a kernel for certifiably secure computer systems being built on the Digital Equipment Corporation PDP-11/45. The design applies a general purpose mathematical model of secure computer systems to an off-the-shelf computer. An overview of the model is given. The paper includes a specification of the design that will be the basis for a rigorous proof of the correspondence between the model and the design. This design and implementation has demonstrated the technical feasibility of the security kernel approach for designing secure computer systems.

Preface

The security kernel design given in this paper is a major revision of a kernel design described in [Schiller]. In the original design a distinction was made between the information and control structures of a computer system, and the access controls dictated by our mathematical model of secure computer systems were only applied to the information structure. To protect the control structure we stated that “it is the responsibility of the system designer to systematically determine all possible channels through the control structure . . . (and prevent) the associated state variable from being controlled and/or observed”. After that design was published it became obvious that the approach to protecting the control structure was not adequate. The systematic determination of channels was equivalent to having a model that protected the control structure.

Consequently, refinements were added to the model to allow the same mechanisms to protect both the information and control structure objects of a system. The basic technique used is to organize all of the data objects in the system into a tree-like hierarchy, and to assign each data and control object explicit security attributes. The major difference between the revised design given in this paper and the original design is the incorporation of the model refinements. In addition, this paper benefits from an additional year’s study and understanding of the computer security problem. Familiarity with the original design is not required.

ware70.pdf

Bibliographic Information

Willis H. Ware, *Security Controls for Computer Systems (U): Report of Defense Science Board Task Force on Computer Security*, The RAND Corporation, Santa Monica, CA (Feb. 1970).

Keywords

secure computing, trap door, Trojan horse, penetration, disclosure, physical security

Abstract or Introduction

With the advent of resource-sharing computer systems that distribute the capabilities and components of the machine configuration among several users or several tasks, a new dimension has been added to the problem of safeguarding computer-resident classified information. The basic problems associated with machine processing of classified information are not new. They have been encountered in the batch-processing mode of operation and, more recently, in the use of remote job-entry systems; the methods used to safeguard information in these systems have, for the most part, been extensions of the traditional manual means of handling classified documents.

The increasingly widespread use of resource-sharing systems has introduced new complexities to the problem. Moreover, the use of such systems has focused attention on the broader issue of using computers, regardless of the configuration, to store and process classified information.

Resource-sharing systems are those that distribute the resources of a computer system (e.g., memory space, arithmetic units, peripheral equipment, channels) among a number of simultaneous users. The term includes systems commonly called time-sharing, multiprogrammed, remote batch, on-line, multi-access, and, where two or more processors share all of the primary memory, multiprocessing. The principle distinction among the systems is whether a user must be present (at a terminal, for example) to interact with his job (time-sharing, on-line, multi-access), or whether the jobs execute autonomously (multiprogrammed, remote batch). Resource-sharing allows many people to use the same complex of computer equipment concurrently. The users are generally, although not necessarily, geographically separated from the central processing equipment and interact with the machine via remote terminals or consoles. Each user’s program is executed in some order and for some period of time, not necessarily to completion. The central processing equipment devotes its resources to servicing users in turn, resuming with each where it left off in the previous processing cycle. Due to the speeds of modern computers, the individual user is rarely aware that he is receiving only a fraction of the system’s attention or that his job is being fragmented into pieces for processing.

Multiprogramming is a technique by which resource-sharing is accomplished. Several jobs are simultaneously resident in the system, each being handled by the various system components so as to maximize efficient utilization of the entire configuration. The operating system[1] switches control from one job to another in such a way that advantage is taken of the machine's most

[1] The system software, which schedules work through the computer system, assigns resources to each job, accounts for resources used, etc.

whit74.pdf

Bibliographic Information

Jerold Whitmore, Andre Bensoussan, Paul Green, Douglas Hunt, Andrew Kobziar, and Jerry Stern, *Design for MULTICS Security Enhancements*, ESD-TR-74-176, Electronic Systems Division, Air Force Systems Command, Hanscom Field, Bedford, MA 01731 (Dec. 1973).

Keywords

MULTICS, containment, access control, operating system secure computing

Abstract or Introduction

The results of a 1973 security study of the Multics Computer System are presented detailing requirements for a new access control mechanism that would allow two levels of classified data to be used simultaneously on a single Multics system. The access control policy was derived from the Department of Defense Information Security Program. The design decisions presented were the basis for subsequent security enhancements to the Multics system.

Preface

This report documents the results of a 1973 study to identify a set of security enhancements for Honeywell's Multics operating system. These enhancements were derived from the Department of Defense Information Security Program. The purpose of these enhancements was to permit users of two different security levels to simultaneously access classified information stored on the Multics system at the Air Force Data Services Center (AFDSC). This report served as a design document for the subsequent implementation of the security enhancements for use at the AFOSC.

The implementation of the design was based upon the "non-malicious" user concept. This concept is predicated upon the assumption that none of the user population would attempt malicious, concerted efforts to circumvent the enhanced security controls. The issues of guaranteeing the impenetrability of the security enhancements were not completely addressed, and the report makes no claim to the system's impenetrability. However, the proposed security controls are thought to be representative of those controls which could be provided on a certifiably secure system. The issues involved in the development of a certifiably secure system are the subject of a separate effort sponsored by the Information Systems Technology Applications Office of the Air Force's Electronic Systems Division.

During the course of the implementation of the security enhancements proposed in this report, several minor design changes were made. This report has not been updated to reflect these changes. This report should be taken neither as a precise

description of the enhanced Multics system implemented for AFOSC nor as a description of Honeywell's Multics Product--current or future.