AD-A060 007 MITRE CORP BEDFORD MASS F/G 17/2
ISSUES IN COMPUTER NETWORK SECURITY.(U)
SEP 78 A C HINCKLEY, J MITCHELL
UNCLASSIFIED MTR-3201 ESD-TR-78-167 F19628-76-C-0001
NL

END
DATE FILMED
12-78
DDC

ESD-TR-78-167

LEVEL (12)

MTR-3201

AD A060007

DDC FILE COPY

ISSUES IN COMPUTER NETWORK SECURITY

BY A. C. HINCKLEY AND J. MITCHELL

SEPTEMBER 1978

Prepared for

DEPUTY FOR TECHNICAL OPERATIONS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
Hanscom Air Force Base, Massachusetts

DDC
RECEIVED
OCT 17 1978
F

Approved for public release;
distribution unlimited.

Project No. 572B
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract No. F19628-76-C-0001

78 10 05 034

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Do not return this copy. Retain or destroy.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved for publication.

William R. Price

WILLIAM R. PRICE, Captain, USAF
Technology Applications Division

C. J. Grewe, Jr.

CHARLES J. GREWE, Jr., Lt Colonel, USAF
Chief, Technology Applications Division

FOR THE COMMANDER

Eric B. Nelson

ERIC B. NELSON, Colonel, USAF
Acting Director, Computer Systems Engineering
Deputy for Technical Operations

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: ESD-TR-78-167	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER: Technical Rept.	
4. TITLE *(and Subtitle)*: ISSUES IN COMPUTER NETWORK SECURITY	5. TYPE OF REPORT & PERIOD COVERED
6. PERFORMING ORG. REPORT NUMBER: MTR-3201	
7. AUTHOR(s): A. C. Hinckley, J. Mitchell	8. CONTRACT OR GRANT NUMBER(s): F19628-76-C-0001
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The MITRE Corp., P. O. Box 208, Bedford, MA 01730	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Project No. 572B
11. CONTROLLING OFFICE NAME AND ADDRESS: Deputy for Technical Operations, Electronic Systems Division, AFSC, Hanscom AFB, MA 01731	12. REPORT DATE: SEPTEMBER 1978
	13. NUMBER OF PAGES: 67
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*: 69p.	15. SECURITY CLASS. *(of this report)*: UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

DDC RECEIVED OCT 17 1978 F

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

COMPUTER SECURITY
MULTILEVEL SECURITY
PACKET SWITCHING
SECURE NETWORKS

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

A secure network, like a secure single-computer system, must incorporate mechanisms for monitoring access to classified objects. In particular, a secure network must protect interprocess communication between hosts. Implementation of a network-wide reference monitor is difficult because information must be transferred from the protected environment of one secure host through the communications network (which will usually incorporate communications processors) to the environment of another secure host. The difficulties are increased because the hosts may be

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

235050

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

20. ABSTRACT (concluded)

operating at different security levels and some may not incorporate multilevel secure capabilities.

Network reference monitor requirements can be met, however, if: a) the reference monitor of each host can obtain the necessary security data to perform access checks for enforcing nondiscretionary and discretionary properties; and b) the transmission between computers (e.g., hosts and communications processors) and between computers and terminals are secured by a combination of physical and logical protection.

Implementation of the network reference monitor may be viewed at different levels — host-to-host, host-to-communications subnet, and communications processor-to-communications processor. In practice, the mechanism may be distributed among these levels and between the different computers originating, transferring and receiving each message.

To make the reference monitor effective, the labeling or identification of the messages is most important. Correct security level and access attributes must be associated with each message or connection, and this information too must be securely transferred between computers. In some cases, user and process authentication data must also be transferred.

Secure transmission paths are assured by a secure communications subnet. Link encryption is necessary to protect the data on the communications circuits. If only link encryption is used, however, the communications processors must meet stringent requirements for multilevel secure software and physical protection.

Multilevel security is necessary for the communications processors. Only a small amount of certified software should be needed as the classified information can be isolated from the communications processor software. If a computer is used to support a Network Control Center operation, close attention must be paid to its security to prevent its use in stealing or corrupting data from the communications processor.

ACKNOWLEDGMENTS

This report has been prepared by The MITRE Corporation under Project No. 572B. The contract is sponsored by the Electronic Systems Division, Air Force Systems Command, Hanscom Air Force Base, Massachusetts.

Many people reviewed this paper during its preparation. John White, Bob Gardella, Kate Peterson, Judah Mogilensky, Dave Snow, and Ed Burke were especially helpful in providing both technical and editorial comments. Any inaccuracies remaining are those of the authors.

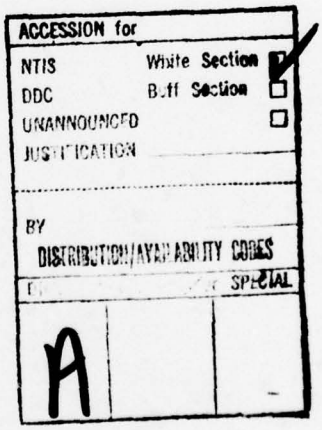


TABLE OF CONTENTS

		Page
LIST OF ILLUSTRATIONS		5
SECTION I	INTRODUCTION	6
SECTION II	BACKGROUND	8
	REQUIREMENTS	8
	SINGLE COMPUTER SYSTEM SECURITY	9
SECTION III	SYSTEMS APPROACH TO NETWORKING SECURITY	13
	NETWORK SYSTEM DEFINITION	13
	NETWORK SYSTEM SECURITY	16
	A Model for Access Checking	17
	Levels of Implementation	19
	SECURITY AND CURRENT NETWORK SYSTEMS	22
	SACDIN	22
	AFSCNET	24
	ARPANET	24
	PWIN	25
	AUTODIN II	25
SECTION IV	SYSTEM LEVEL ISSUES	26
	ACCESS CONTROL	26
	Located at the Origin Computer	27
	Located at the Destination Computer	28
	Located at Both the Origin and Destination	28
	Located at a Centralized Computer	29
	Conclusion: Sorting Out the Options	29
	IDENTIFICATION	31
	Identification on Messages	31
	Authentication	33
	AUDITING	36

PRECEDING PAGE NOT FILMED
BLANK

TABLE OF CONTENTS (Concluded)

		Page
SECTION V	SUBSYSTEM LEVEL ISSUES	38
	COMMUNICATIONS	38
	Background	38
	Circuit Protection - Encryption Issues	39
	Communications Processor Security	44
	Control System	48
	Limited-Access Controls	52
	Broadcast System Issues	53
	Summary	56
	USER STATIONS	57
	Secure Office Terminal Program	57
	Secure Office Terminal - Issues	57
	Summary	59
	NETWORK FRONT-END PROCESSORS	59
	SECURITY OFFICERS	61
SECTION VI	SUMMARY AND RECOMMENDATIONS	62
	MAJOR NETWORK SECURITY ISSUES	62
	Distribution of Control Mechanisms	62
	Identification and Authentication	62
	Encryption Techniques	63
	Communications Processors	63
	Network Front-End Processors	63
	Network Useability	63
	RECOMMENDATIONS	63
REFERENCES		65

LIST OF ILLUSTRATIONS

Figure Number		Page
1	Reference Monitor	10
2	Illustration of Terminology in a Typical Configuration of a Network System	14
3	Examples of Network Topologies	15
4	Two Ways of Viewing Access in an Information Transfer	18
5	A More Detailed View of Access in an Information Transfer	18
6	Access Checking at Host-to-Host Level	20
7	Access Checking at Host-to-Communications Processor Level	22
8	Access Checking at Subnet Level	23
9	Typical Blacker Configuration	43
10	Proposed Security Protection Module (SPM)	45

SECTION I

INTRODUCTION

In recent years computer networks have advanced from academically interesting research projects to functionally necessary systems in military as well as civilian applications. The Air Force is participating in this advance through the development of SACDIN and the Systems Command's Network (AFSCNET). Increasing impact on Air Force operations will result from its participation in the Prototype WWMCCS Intercomputer Network (PWIN), the implementation of DCA's AUTODIN II, and the development of the operational WWMCCS Intercomputer Network (WIN) and the Joint Tactical Information Distribution System (JTIDS).

With the application of networks to military needs, security has become a critical issue, as it did when single computer systems were used first in military applications. A network is more difficult to secure than a single computer, however, since the network's components may be dispersed and controlled by different managements.

Commenting on both the broader exposure and potentially greater vulnerability, Schell and Karger have pointed out that "networks can have a major adverse security impact by:

"1) dramatically increasing the number of users with potential unauthorized access;

"2) potentially making the security controls on a specific host irrelevant by making information accessible to other hosts that do not have effective security controls; and

"3) introducing additional vulnerabilities through the lack of effective security controls in network elements, e.g., insecure network communications processors" (1).

Many documents have been written on various aspects of network security. Most advocate the well-established principle that security must be designed into systems at their inception. That is, retrofitting or patching does not provide effective protection for a network. This paper identifies the issues that need to be analyzed in designing secure networks. Its purpose is to encourage comments and analysis by other interested agencies and users.

The investigation of network security issues is based on the ESD/MITRE program in computer security as outlined in Section II. A network system viewpoint is adopted in Section III in order to define the important components and establish a model of network security to serve as a framework for detailed discussions. Some security issues - primarily those of access control, identification, and authentication - pervade all aspects of network design. These are discussed in Section IV. Section V outlines security problems related to certain network subsystems, primarily the communications subnet. We conclude by summarizing the major issues and suggesting work programs to resolve them in Section VI.

SECTION II

BACKGROUND

REQUIREMENTS

A major problem in military applications of ADP systems is the shared use of resources having different classifications and formal category sets by users having different security clearances and formal category sets. We shall use the general term security level for both users and resources to denote the combination of:

1) clearance or classification; and

2) formal compartment or category set.

The objectives of a computer security system are to prevent compromise of classified information, to prevent unauthorized modification or insertion of data, and to prevent an intruder from denying service to an authorized user. By controlling access to classified information in accordance with appropriate security levels, compromise can be prevented and important contributions can be made to improving the integrity of computer systems. The enforcement of access control based on security levels constitutes formal or nondiscretionary security. It is also necessary to enforce discretionary or need-to-know requirements.

Current security procedures as defined in DoD Directive 5200.28 address these requirements by either:

1) clearing all users to the highest level of information on the system and processing all work at that level; or

2) processing jobs of different levels at different times thereby requiring a complete system change or sanitization (color change) each time the level is changed.

Under either of these procedures, all simultaneously operating processes are at the same security level. Hence, we term such a system unilevel secure. The operation of such a system is usually termed 'system high'. Unilevel security is costly and in some applications not operationally possible.

Hence, a system is needed which automatically enforces nondiscretionary and discretionary security. Ideally, such a multilevel system would be openly available for uncleared as well as

cleared users. If the open environment were too threatening, however, a closed multilevel system with no uncleared users could be designed instead.

A mechanism to enforce security in an open multilevel system should not make the system unuseable. Although there must be procedures to allow the user to be aware of the classification of the material he uses and to allow him to be confident that he will not jeopardize the security of this material, the procedures must not overburden him. If they do, he will tend to ignore the security constraints and/or his productivity on the system may be reduced. Therefore, ease of use is an important requirement of any system security design.

SINGLE COMPUTER SYSTEM SECURITY

In 1972, the Anderson Panel (2) analyzed the problem of vulnerability in an open, multilevel, single-computer system. The panel was convened after personnel from the Electronic Systems Division (ESD) and MITRE determined that there was no set of modifications that would secure GCOS III for open multilevel operation at the Air Force Data Service Center. (GCOS III is the operating system for the Honeywell 635 and 6000 computer.) The panel recommended as a technical approach "to start with a statement of an ideal system, a model, and to refine and move the statement through various levels of design into the mechanisms that implement the model system" (2).

The basic component of the ideal system proposed by the security technology panel is the reference monitor - an abstract mechanism that controls access of subjects (active system elements) to objects (other system elements) within the computer system. Figure 1 illustrates the relationships among the subjects, objects, reference monitor, and reference monitor authorization data base.

An implementation of the reference monitor abstraction is called a reference validation mechanism. It permits or prevents access by subjects to objects, making its decision on the basis of subject identity, object identity, and security parameters of the subject and object. The implementation both mechanizes the access rules of the military security system and assures that they are enforced within the computer.

To be effective, the reference validation mechanism must be designed to meet the following three requirements.

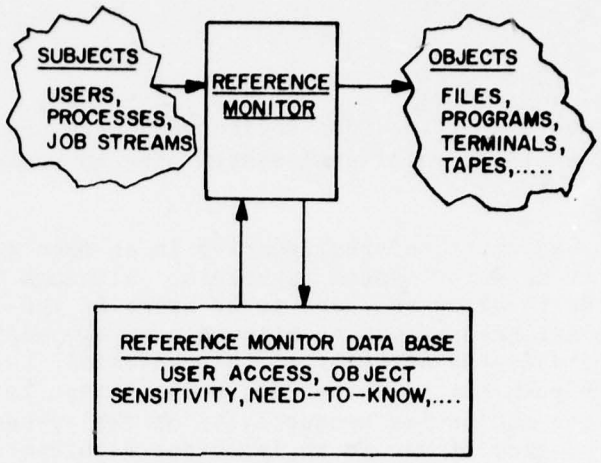


Figure 1. REFERENCE MONITOR

1. Completeness - The mechanism must be invoked on every access by a subject to an object.

2. Isolation - The mechanism and its data base must be protected from unauthorized alteration.

3. Verifiability - The mechanism must be small, simple, and understandable so that it can be completely tested and verified to perform its functions properly.

The combination of hardware and software required to meet these criteria will be called the security kernel.

To date, ESD/MITRE and others have focused their ADP system security research primarily on single-computer systems. Recognizing the Anderson panel's "ideal model" as an important starting point, ESD initiated development of a mathematical model of computer security in 1972. The completed model (3) represents a secure computer system as a finite-state mechanism that makes explicit transitions from one secure state to the next. The rules of the model, which formally define the conditions under which a transition from state

to state can occur, are proven to allow only transitions that preserve the security of information in the system.

There are two basic properties in the model. The first, the simple security property, is satisfied if the security level of any subject observing (reading) an object dominates the security level of the object. (One security level, L1, dominates another, L2, if the clearance or classification of L1 is greater than or equal to the clearance or classification of L2, and the category set of L1 includes the category set of L2.) That is, a subject may only read information at a level less than or equal to its own.

The second property, the *-property (pronounced star property), restricts all but proven and, therefore, trusted subjects from writing information at a lower level than the maximum they read. In a manual system this responsibility is assumed by each user who is trusted to avoid illegally downgrading classified information to which he has access. More formally, the *-property is satisfied if, for any subject having simultaneous observe access to object-1 and alter access to object-2, the security level of object-1 is dominated by the security level of object-2.

Implementing these two properties ensures formal or nondiscretionary security. A third property of the model, the discretionary security property (ds-property), ensures that discretionary security will be provided when the model is implemented. In the model, the discretionary property depends on a matrix, M, whose rows represent subjects and whose columns represent objects. The intersection of a row and column contains the access attributes (observe or alter) for that subject and object. Formally, the ds-property can then be defined as requiring that a subject, i, can only observe or alter an object, j, if that access attribute is in the i,jth component of the matrix, M.

Together, the simple security, *-, and discretionary security properties provide specific requirements for the reference monitor and the security of the system. Once the model is shown to uphold the Department of Defense regulations, validating the security of the system is reduced to providing complete assurance that the reference monitor behaves as the model requires. A formal administrative certification is appropriate after validation.

Although the Anderson Panel was concerned with the security of single-computer systems, the results are applicable to network security since they provide a definition, an approach, and a common concern for ADP system security. In particular, networks also

require a reference validation mechanism to check access of all subjects to all objects. As in a single computer, the network reference mechanism must be complete, isolated, and verifiable. The following section discusses network systems and how the access control concepts developed for the single-computer system might be applied to networks.

SECTION III

SYSTEMS APPROACH TO NETWORKING SECURITY

NETWORK SYSTEM DEFINITION

Although computer networks evolved out of the union of communications and computer science, their security issues are more than the simple union of the security issues of communications and computers. The problems in both areas combine and new problems emerge from the interactions between multiple computers and communications lines. Therefore, we have adopted a systems viewpoint that encompasses the various network components, their interrelationships, and their interdependencies.

Computer network systems have been described with many different terminologies from the related disciplines. Generally, we shall use standard terminology defined by the National Bureau of Standards (4) and by Cotton and Benoit (5). A network system is composed (see Figure 2) of a set of nodes, a set of communications lines connecting the nodes, and a set of protocols (rules) specifying how the nodes should communicate over the lines. We shall examine each of these components in defining a Network System below.

A node incorporates a communications processor (or switch) and at least one (not necessarily distinct) host processor serving the network users. At one extreme, the same processor may serve both host and communications functions (e.g, in the ARPANET Terminal Interface Processor). At the other extreme, the host and communications units may each consist of a multiprocessor. A typical node configuration may include a communications processor, a host, and a network front-end-processor (NFEP) to perform pre- and post-processing for the host.

The lines form a topology that is either centralized (i.e., hierarchical, tree or, in a simple case, star) or distributed, as illustrated in Figure 3. For the purposes of this analysis, the distributed topology will be used as the primary example because it involves complex security problems and is often a design choice for nationwide networks currently in use or in development. Most of the issues and their discussion, however, also apply to the centralized networks.

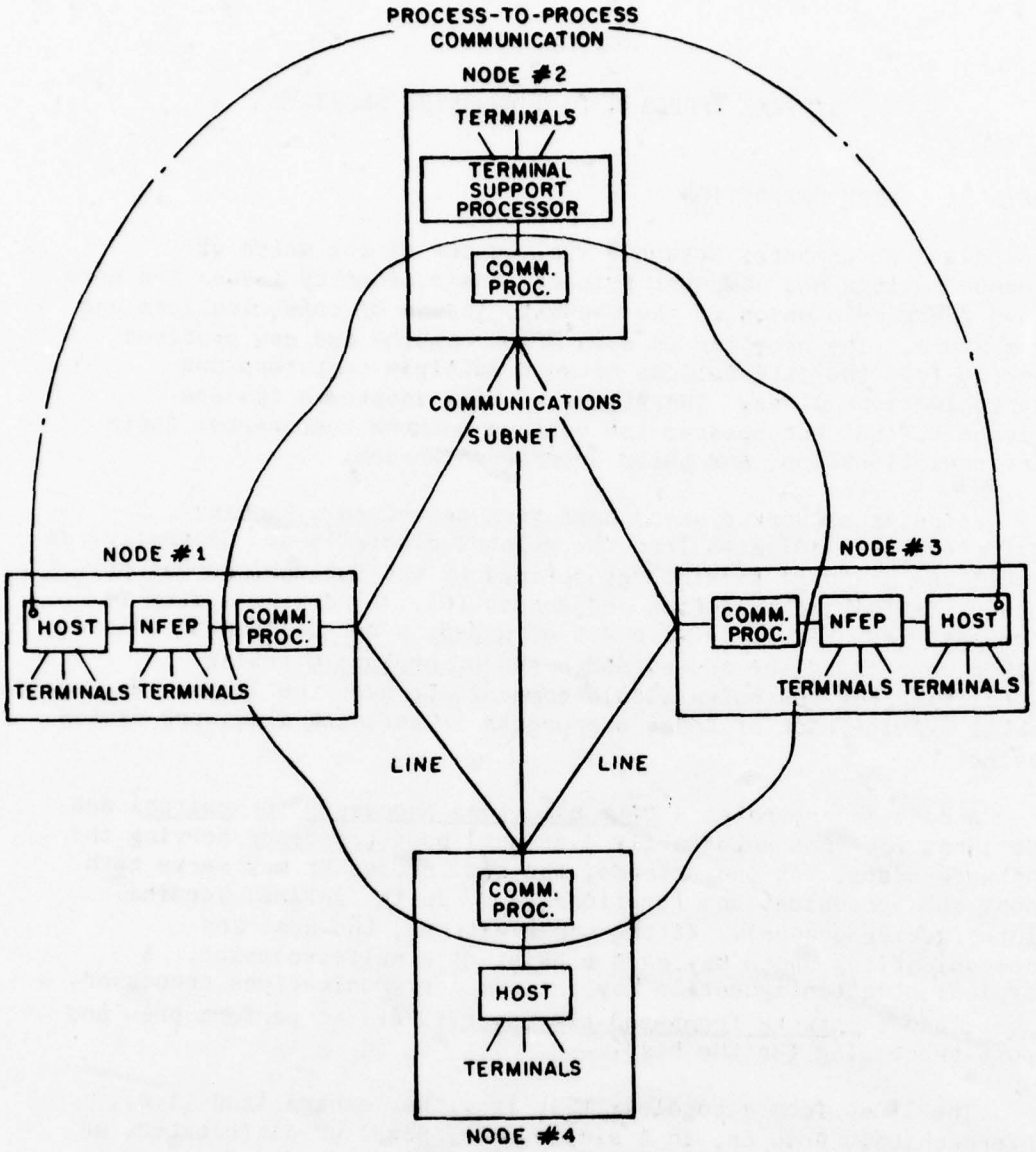
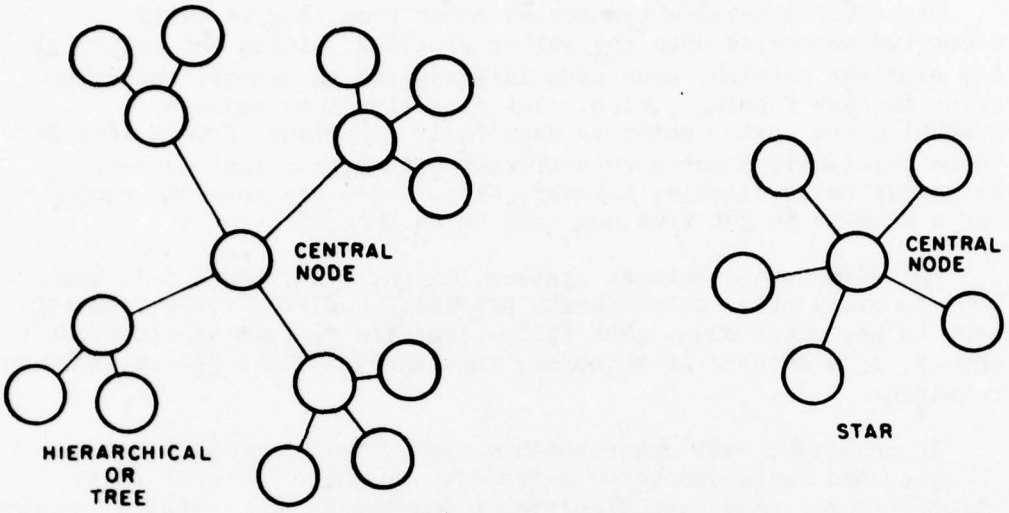
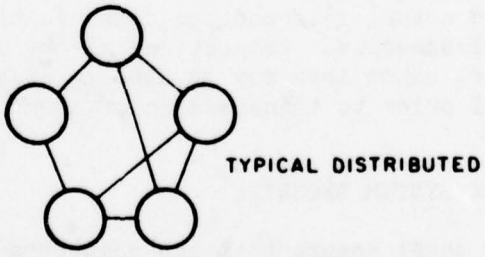


Figure 2. ILLUSTRATION OF TERMINOLOGY IN A TYPICAL CONFIGURATION OF A NETWORK SYSTEM



(a) CENTRALIZED TOPOLOGIES



IA-47,734

(b) DISTRIBUTED TOPOLOGIES

Figure 3. EXAMPLES OF NETWORK TOPOLOGIES

78 10 05 034

Distributed network topologies range from ring to fully-connected depending upon the number of lines joining the nodes. In the simplest network, each node is connected to exactly two other nodes thereby forming a ring. The ring structure reduces communication costs, which is especially important if each line is to be physically secured or outfitted with encryption devices. Rings may be unreliable, however, since there are only two routes for a message to get from one node to another (6).

Fully-connected network systems, on the other hand, join each node to every other node thereby providing a direct route from any node to any other node. But fully-connected systems are enormously costly; in a network of 40 nodes, for example, 780 circuits would be required.

In practice, most distributed networks fall somewhere between the loop and fully-connected extremes. We shall focus on these networks since they have significant complexity and, usually, enough nodes to preclude the expensive fully-connected topology.

The protocols in a network system facilitate interprocess communication - the primary service in a network. Protocols for process-to-process communication can be decomposed into: host-to-host, host-to-communications processor, and communications processor-to-communications processor protocols. In some networks, one of the primary functions of the host-to-host protocol is to make and break connections between processes.

A connection consists of a path or logical link between two computers with a process (and an associated buffer) at each end. Constructed by an exchange of information between the hosts before any communication occurs, a connection ensures that a destination process can be addressed correctly. Cerf and Kahn (7) suggest that connections may be superfluous if the network is configured to allow all necessary addressing and sequencing information to be exchanged via the actual interprocess communications and their acknowledgements. Connections may be useful for security purposes, however, since they may be used to exchange data needed for access control prior to transmission of communications.

NETWORK SYSTEM SECURITY

We shall assume that the computers in a network are each autonomously secure. That is, they may have a reference monitor to implement multilevel security or they may be physically secured and

confined to operate at a single security level (including unclassified). The goal in a secure network system is to link these secure computers together in a manner that allows network-wide interprocess communications without introducing any formal or discretionary security violations.

Enforcement of formal and discretionary security for network-wide interprocess communication requires both:

1) an access check for each communication between the processes; and

2) secure transmission of the communication over a path between the hosts.

We shall develop a conceptual model for access checking in the next sub-section. Secure transmission requires a secure communications subnet, requirements for which are addressed in Section V.

The conceptual model presented here serves as a basis for discussing issues in the protection of classified information. In the future, the model should be extended in two directions.

1) Correct and reliable transfer of important information must be assured.

2) The conceptual model should, if possible, be developed into a mathematical one. The presence, in a network, of simultaneous, asynchronously operating and interacting processes may make this difficult. Even more difficult will be the task of verifying that a particular network conforms to the model.

A Model for Access Checking

As shown in Figure 4, a transfer of information from a sending process to a receiving process can be viewed as either:

1) the sending subject writing information to the receiving object; or equivalently

2) the receiving subject reading information from the sending object.

In reality, a subject in a network does not actually read from or write to a remote process directly. Rather, the sending process

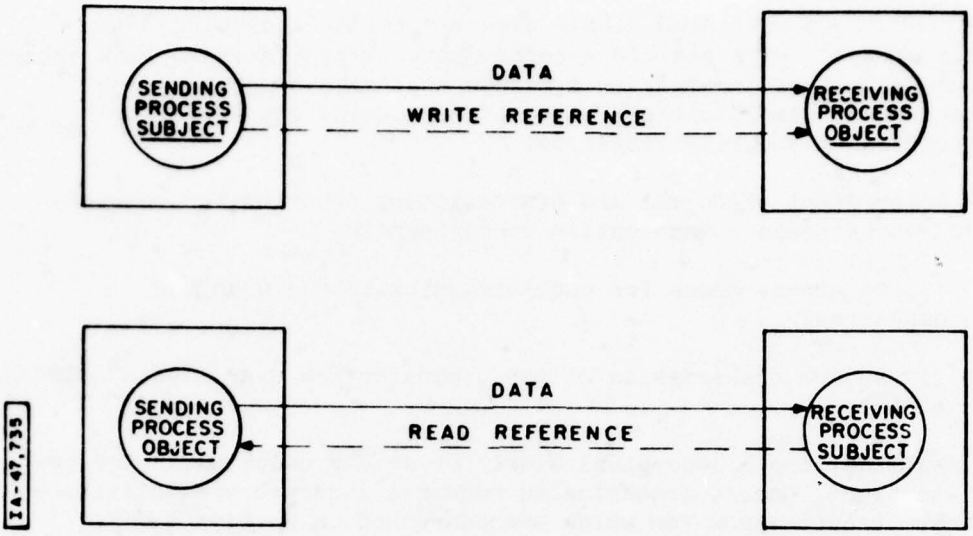


Figure 4. TWO WAYS OF VIEWING ACCESS IN AN INFORMATION TRANSFER

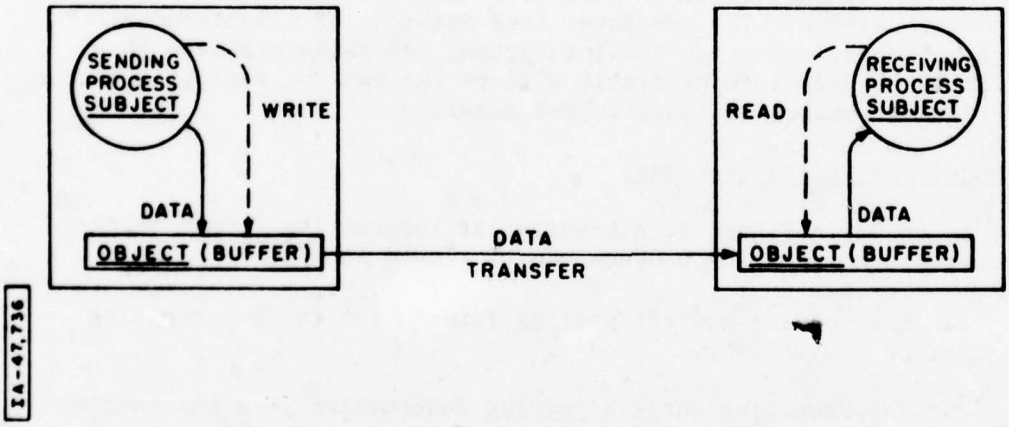


Figure 5. A MORE DETAILED VIEW OF ACCESS IN AN INFORMATION TRANSFER

writes into a buffer, the buffer is transferred by hardware and software to a buffer at the remote site, and the receiving process reads the information from the buffer. Figure 5 shows the subject/object relationships if the information transfer is viewed in this manner.

Suppose the buffer in the sending host had the same security level as the receiving process. Then a reference monitor's test of the write operation (actually an append) into that buffer would constitute a formal access check for the information transfer. Alternatively, suppose the buffer in the receiving host had the same security level as the sending process. Then a reference monitor's test of the reading of that buffer by the receiving process would constitute a formal access check for the information transfer.

In Section IV, these two models of access control in interprocess communication are used to examine access checking and the tradeoffs involved in choosing where the reference validation mechanism should be located. Because the reference monitor must check every access, some portion of the mechanism must be distributed. The choices for the policy enforcement are:

1) in the sending host;
2) in the receiving host;
3) in both the sending and receiving hosts; and
4) in another centralized network node.

Assignment of security levels to the buffers used in the model requires:

a) the transfer of security level data between hosts; and
b) a procedure to make the assignment.

If multiple transmissions are sent between the same two processes, repeated transfer and assignment of buffer security levels is both time consuming and a cause of excess network traffic. Use of a process-to-process connection can eliminate this redundant activity. When the connection is made, the buffers would be assigned security levels and access attributes to be retained throughout a sequence of communications. Additional security level information transfers and buffer assignments would not be needed for each transmission.

Levels of Implementation

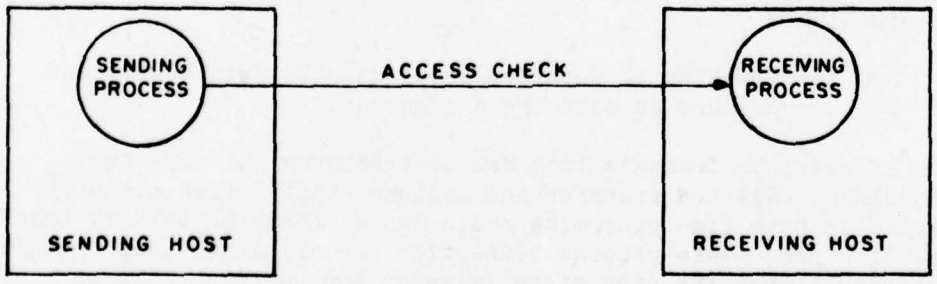
The reference monitor functions for access control in network-wide interprocess communication may be implemented at different levels: host-to-host; host-to-communications processor; or

communications processor-to-communications processor. Of course, the reference monitor must have accurate access control information to function correctly. This subsection identifies major issues in the implementation level; in Section IV the problems of ensuring correct identification are discussed.

Host-to-Host Level

Figure 6 illustrates the necessary reference monitor functions if access control is viewed at the host-to-host level. Only one access check is performed for each transmission and it ensures that the security level of the sending process is less than or equal to the security level of the receiving process.

Access control performed only at this implementation level also requires that the path from the sending host to the receiving host be secure. The path might be made secure if, for example, end-to-end encryption were used to avoid exposing any classified information to the lines and communications processors. To function correctly, an access mechanism at this implementation level also requires that the path transfer the access control information accurately.



IA-47,737

Figure 6 ACCESS CHECKING AT HOST-TO-HOST LEVEL

Host-to-Communications Processor Level

A host-to-communications processor implementation requires access checking as the communication is transferred 1) from the sending host to the source communications processor, 2) from the source communications processor to the destination communications processor, and 3) from the destination communications processor to the receiving host. At each interface, the level of the origin process must be less than or equal to the level of the destination process.

Figure 7 illustrates the access checks which, if positive, constitute a reference check for the complete interprocess communication between host processes. This check is based on transitivity, which says: if information is allowed to go from process-1 to process-2, and from process-2 to process-3, then it may go from process-1 to process-3.

This level of implementation assumes that a) the physical links between hosts and communications processors are secure (e.g., by encryption), and b) the path from the source to the destination communications processor is secure. If the communications subnet is secure and interprocess communication between the source and destination communications processors can be trusted, then the access check between the two communications processors is implicit.

Communications Subnet Level

It may be desirable to perform access checks as a communication moves between each communications processor in the subnet. Coupled with the checks upon entering and leaving the subnet, an access check would then be made for each transmission from one computer to another between the sender and receiver.

Figure 8 illustrates the access checks required at this level. If all the subnet access checks are positive, then by transitivity the access check between the source and destination communications processor is positive. That composite access check, combined with the host to subnet interface access checks, makes up the required host-level access check.

Implementing the reference monitor functions at this level requires that the lines between the components be secure. Link encryption can provide such protection.

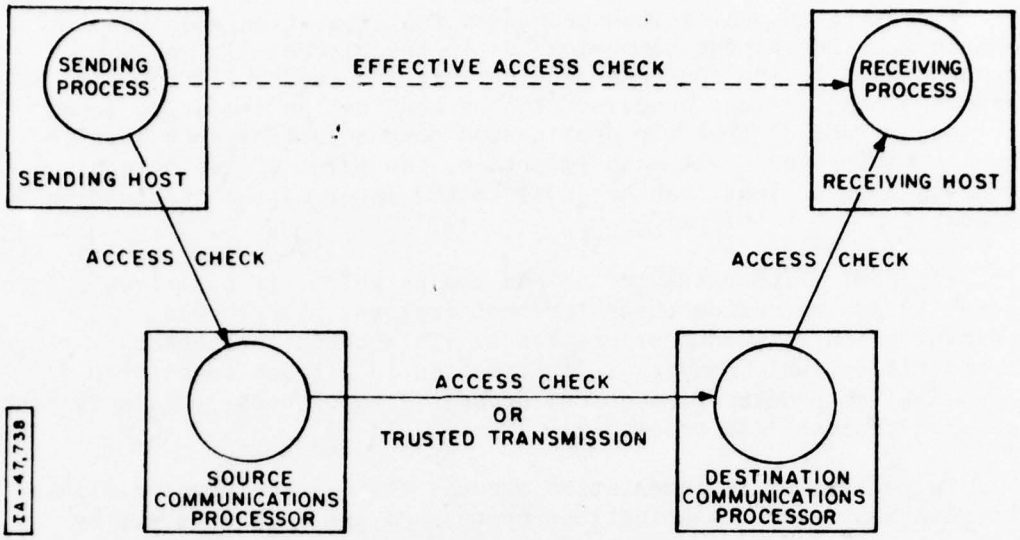


Figure 7. ACCESS CHECKING AT HOST-TO-COMMUNICATIONS PROCESSOR LEVEL

SECURITY AND CURRENT NETWORK SYSTEMS

As mentioned above, networks have become integral parts of various military programs. The Air Force, for example, is working on SACDIN, PWIN, and AFSCNET among others. AUTODIN II is being developed by the Defense Communications Agency (DCA) which is also responsible for the management of the ARPANET.

SACDIN

SACDIN, the Strategic Air Command Digital Network, is a packet-switching system that will support the World Wide Record/Data Command Control Communications requirements of SAC and the National Command Authorities. It will have no resource-sharing capabilities since it is primarily a message communication system.

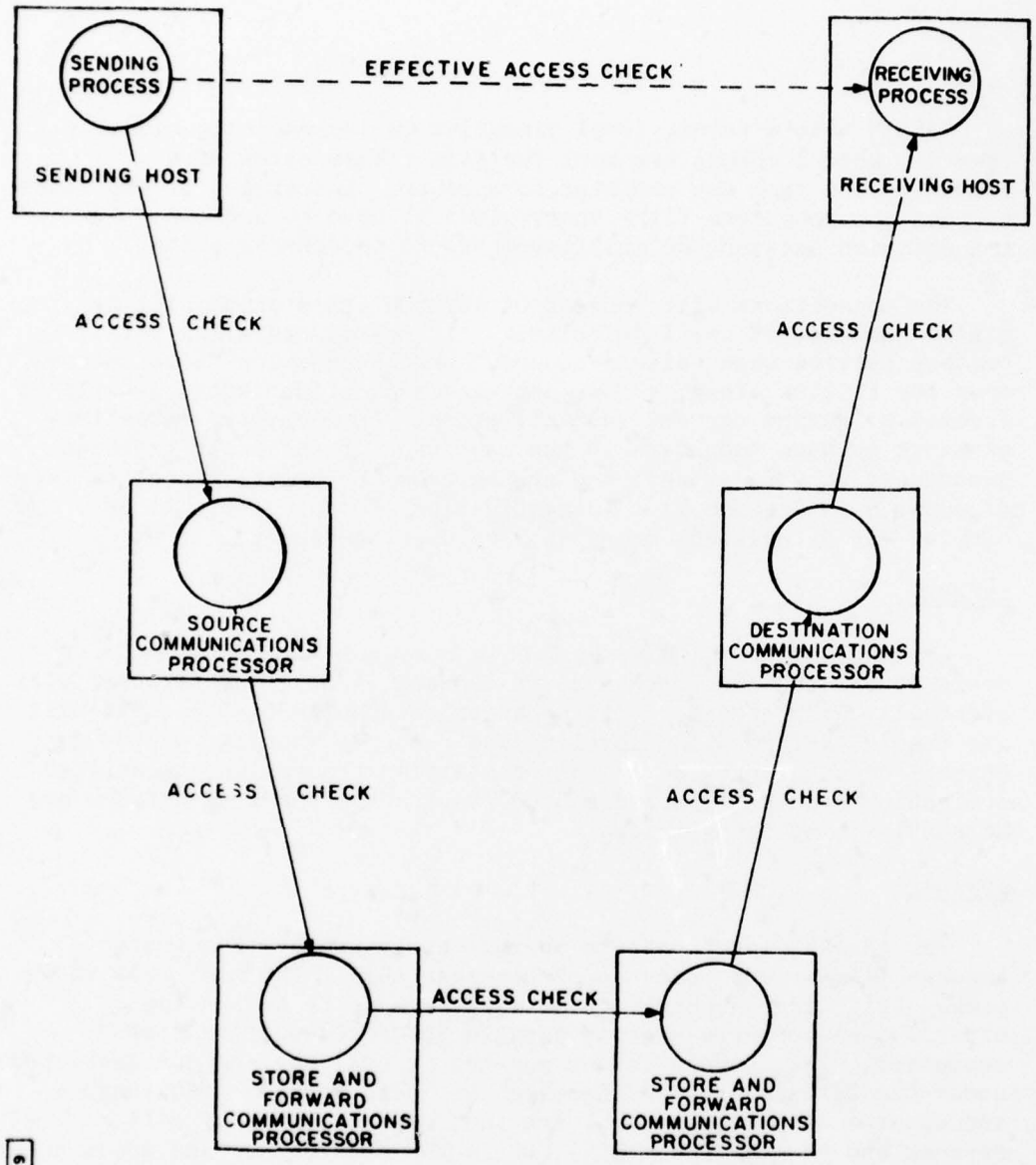


Figure 8 ACCESS CHECKING AT SUBNET LEVEL

SACDIN uses a subnet level viewpoint in implementing security. That is, access checks are made for each transmission of a communication from one computer to another. Encryption of the lines between the computers (link encryption) is used to protect transmission between the multilevel secure processing units.

The connections will consist of AUTOVON lines which will be dialed and kept in use indefinitely. Automatic redialing will restore service when failure occurs. When connection first occurs over the AUTOVON lines, a four-part authentication technique will be invoked to ensure correct identification. An encrypted Date-Time exchange between computers is the last part of the authentication procedure; it will protect the system from the playback problem (see discussion in Section V). Authentication of the users will be carried out by physical means at each user terminal.

AFSCNET

AFSCNET, unlike SACDIN, will be a resource-sharing network. Designed for the Air Force Systems Command (AFSC), the network will eventually link together sixteen heterogeneous AFSC sites. It will use the ARPANET for load leveling and resource sharing. Although Phase I does not provide for any classified processing, classified processing requirements and procedures for the Phase II network are currently being investigated.

ARPANET

The ARPANET interconnects approximately eighty-five hosts through fifty-eight nodes as of December 1975. The most well known packet-switching network, its primary purpose is to provide experimental resource sharing among a heterogeneous group of computers. The network is now managed by DCA, having been developed under the Defense Advanced Research Projects Agency. Maintenance and updates of the subnetwork are the responsibility of Bolt, Beranek and Newman, Inc. (BBN), while the maintenance and updates of the individual host sites are the site's own responsibilities.

Security was not a design consideration in the development of the ARPANET. Recently, however, BBN has described (8) the Private Line Interface (PLI) which will provide end-to-end encryption for network connections. It will also provide insertion or deletion of network protocol information between the communications processor (called the IMP) and a bit stream source or sink.

PWIN

PWIN is the prototype World Wide Military Command and Control System's (WWMCCS) Intercomputer Network sponsored by the Joint Technical Support Activity of DCA. It will connect Honeywell 6000 series computers at various WWMCCS sites. Eventually, PWIN may include other computers and use the AUTODIN II communications system.

PWIN makes a clear separation between hosts and the communications subnet. Therefore, it is not surprising that the designers adopt a host viewpoint towards network-wide access checking. Although development is still underway, it appears that link encryption and the National Security Agency's (NSA) crypto technology called BLACKER may be used. Essentially, there are two basic components located at each node; one controls encryption and the other access. This distributed configuration is to be distinguished from an earlier technology that used an Intelligent Crypto Device (ICD) and a centralized Security Controller (SC) (9) to handle the access control and some of the cryptographic functions. Both technologies are discussed below in Section V.

AUTODIN II

AUTODIN II, also proposed by DCA, will be another packet-switching network to provide a common communications system for computer and terminal intercommunication. It will consist initially of eight switching centers, each connected to almost every other center.

The AUTODIN II designers, like the PWIN designers, draw a clear distinction between the hosts and the communications subnet. Although the level at which access control is to be implemented is not clear, the hosts appear to be responsible for security. But AUTODIN II, essentially a communications subnetwork, will check security levels of messages entering and leaving the subsystem. Therefore, security may be envisioned at the subnet entry/exit level in this network.

Encryption will be provided for all links and, in a few limited cases, for end-to-end communication. The communications processors are to be thoroughly tested and redundant security level labels will be used in an effort to ensure the label's correctness.

SECTION IV

SYSTEM LEVEL ISSUES

In this section we discuss the issues of access control, identification, and auditing as they affect the design of a network system. Alternative resolutions of these issues are shown to place certain requirements on individual subsystems of a network, especially the communications subnet. In Section V we discuss the subsystems and how they may be designed to meet these requirements.

ACCESS CONTROL

In Section III we developed a model (Figure 4) for viewing access control between a sending process in one host and a receiving process in another host. Although the model was presented for the access check in a host-to-host level implementation, it applies in general to any access check between two processes at any level of implementation.

Therefore, the following discussion examines the model in a generalized setting, concentrating on the four choices for locating the reference monitor functions for an arbitrary access check between processes in different hosts. For each of the choices, there are two important issues:

a) how and when should the necessary security data (i.e., security level and access attributes) be transferred between hosts; and

b) what assumptions of trust would have to be made about the hosts and the path between them.

Clearly, the security data must be transferred accurately (as well as securely) if the access control is to be effective; this problem requires correct identification as discussed in the subsection on identification.

In this discussion, we shall use the term **message** to refer to any communication, segment, or packet. Furthermore, we shall always refer to one access check even though the message may be packetized and sent in parts. The access checks for each packet would constitute a check for the entire message.

Located at the Origin Computer

If the reference monitor functions were located at the origin computer, the data describing the security level of the destination process would have to be transferred to the origin computer. The distribution of security level data for access control may occur in two ways. Each computer might repeatedly transmit information about the existence and security level of its processes so that the other computers could maintain this data in their own files. Alternatively, an origin computer could request from the destination computer the specific data needed for each decision.

The first distribution option is only realistic in a network in which there are only a few processes and these change infrequently, since otherwise the network traffic and the computer data bases would be flooded. The second option may be unacceptable if time is critical, since each message would require two other messages to authorize its transmission.

Once available at the origin computer, the remote process's security level must be stored for use by the reference monitor. In the model illustrated in Figure 4, this could be achieved if the origin buffer were assigned the same security level as the remote process. Then the access check for the origin process's append operation into that buffer would accomplish the required formal access check for the interprocess communication.

After a secure transmission to the destination host, the message would be left in a buffer also having the security level of the destination process. This level, determined and assigned by the destination host, does not rely on security data from the origin computer. The access check as the destination process reads the buffer will, of course, be successful.

Discretionary access attributes are granted by an origin process for a particular message. The attributes are used when other processes attempt to access the message. Under the assumption that access checking is at the origin computer, remote processes can only achieve access by a complex process such as executing a surrogate process in the origin computer. A secure delivery to the destination process can be assumed, however, if the destination host assigns access attributes to the destination buffer so that only the destination process can read the message.

Locating reference monitor functions solely in the origin processor may be unacceptable since the destination computer must accept every message blindly. The destination computer must trust that another computer has performed a proper access check, and that the identification of the destination process on the message (to be used to retrieve the level of the destination process and to create the default access attributes for the buffer) is correct. That is why the destination computer can do no more than set up a buffer with a default security level and default access attributes.

Furthermore, a process, once initiated, must never change its security level. Otherwise, a message, having passed an access check for the previous security level, might arrive at the destination process with an illegal security level relative to that process.

Now consider the effect if either computer were not multilevel secure. If only the origin computer were unilevel, access checking could not exist there because there would be no reference monitor. Therefore, no unilevel secure computers, in a network in which access is checked solely at the origin computer, can send messages to multilevel hosts. Alternatively, if only the destination computer were unilevel, all processes on it would have the same security level. In this case, all subjects in the destination host could access any message. Finally, if both the origin and the destination computers are unilevel secure, they must always have the same security level, as must every other unilevel computer to which they send messages. An external mechanism (e.g., the communications processors) may be used to ensure that unilevel computers only send and receive messages at their operating security level.

Located at the Destination Computer

Even if access checking is located in the destination computer, knowledge about that computer must still reside in the originating computer. The originator must know the highest level of the destination, and the mode (either single or multilevel). The originator can only send messages at the appropriate single level to a single level destination. Even a multilevel destination may have a maximum level below that of the originating message. Thus it is not feasible to have all controls reside in the destination computer.

Located at Both the Origin and Destination

Complete access checking at both the origin and destination computers would require security data to be passed between the hosts. Of more value is the combination of partial and complete access checking at both the origin and destination which could draw on the

benefits of multiple access checks without incurring too great an overhead.

For example, a formal access check for the maximum security level of the destination computer could occur at the origin, preventing messages being sent to a host of lower security. The expense would be small, since the origin's data base of computer security levels would require infrequent updating. Then a complete access check at the destination could enforce discretionary as well as formal security. Such multiple access checks would eliminate blind trust that: a) a message will arrive at its proper destination after a check at the origin, or b) that a message at the destination has already been checked somewhere else.

Located at a Centralized Computer

If the reference monitor for all messages were in a central computer somewhere in the network, security data for all subjects and objects would have to be transferred there. If this information were constantly being sent to the central node, a large volume of message traffic would be created for this reason alone. On the other hand, individual requests for the security data for each message might cause large time delays. Therefore, centralized access checking appears reasonable only in a host-to-host level implementation that uses connections or in a star network. This concept has been discussed by Branstad (18) whose "Agent" responds to requests for connection between two processes by testing the access rights. If the connection is acceptable, it is implemented by distribution of crypto keys to the participating computers.

A major problem with centralization is the reliability of the system. If the node containing the reference validation mechanism were to fail, all network interprocess communication would be aborted. This weakness could be alleviated with redundant or regional access mechanisms.

Conclusion: Sorting Out the Options

Analysis of the options for choosing an implementation level and access check locations might proceed as follows:

1) An access control mechanism at the host-to-host level may be required if the communications processors are not multilevel secure.

a) If all the hosts are multilevel secure to the same maximum level, access checking could occur at either the sending, receiving, or both hosts. It would be sufficient and preferable, however, to choose the receiving host.

b) If the hosts are multilevel secure to different maximum levels, or some of the hosts are only unilevel secure, an additional check at the sending host (of the destination host's current security level) would be necessary.

c) Unilevel secure hosts may communicate with other unilevel hosts only if both computers are at the same level. This restriction would have to be enforced at the communications subnet entry and exit points.

2) If access control is implemented at the host-to-communications processor level, the source and destination communications processors may be used to police messages entering and leaving the hosts.

a) If all the communications processors are multilevel secure to the same maximum level, both host-to-communications processor access checks (at entry to and exit from the subnet) should be implemented in the communications processor.

b) If some of the source and destination communications processors are multilevel secure to different maximum levels, or some are only unilevel secure, access checking would be needed between the source and destination communications processors. Then access should be viewed at the communications subnet level.

3) Checking access at the subnet level would be necessary if a configuration allowed both source and destination communications processors to be unilevel. The communications processors would police messages to and from the source and destination. Implementations at this subnet level may be more desirable since the trusted paths are only physical links.

a) If all the communications processors were multilevel secure to the same level, access checking between them could always occur at the destination.

b) If any of the communications processors were multilevel secure to different levels, or some were unilevel, access checking between all communications processors would have to occur at both origin and destination.

In summary, whichever level is chosen will require that the access checking mechanism be distributed among the computers as determined by the degree of security at each computer. The access control data base should be located at the computer performing the

access check to minimize extra transfer of access data over the networks.

IDENTIFICATION

Identification of a subject or object in a network includes:

1) its unique name;

2) its security level; and

3) its access attributes.

We see two distinct issues pertaining to identification in networks. First, identification placed on messages for network-wide interprocess communication must be correct for the proper functioning of the access control. Second, identification throughout the network must be authentic. The identifications supplied by permanently connected components will be authentic, but an authentication mechanism must establish the identity of components which may be separated from the network each time they are reconnected. In the next two discussions, we examine these issues and their effect on network design.

Identification on Messages

The messages transferred between processes identify a subset of the following:

1) the origin process;

2) the destination process; and

3) the message itself.

The exact choice of which identification parts are placed on a message depends on the configuration. In particular, the implementation level and the location of the access checking influence both which identifications are needed and when they are used.

Problems with identification on messages fall into three categories: creation, assignment, and transmission. Unauthorized persons or processes must not be able to insert, delete, or modify identifications. The implications of these requirements,

particularly as they pertain to the access control mechanism, are developed in the following discussions.

Creation

When an identification is created, the name must be unique and the security level and access attributes must be correct. Unique names are necessary if messages are to arrive at their correct destinations, the network is to acknowledge the proper messages, and the access control mechanism is to retrieve the access attributes for discretionary security. To ensure that a name is unique throughout the network, the computer name may be appended to the process name for all intercomputer messages. Alternatively, a network may use a unique network-wide notation into which each host must map its local names. In the ARPANET, for example, sockets serve this purpose.

Correct security levels and access attributes are necessary if an access control mechanism is to enforce the DoD formal security policies. If the identification is created in a computer operating with system-high security, the security level is that of the operating level of the computer, but an external mechanism (e.g., the communications subnet) must ensure that the label is created correctly.

Assignment

Assuming the sender's, receiver's and message's identifications are created correctly, they must be assigned correctly to messages as necessary. If the origin computer is multilevel secure, trusted I/O software can make the assignments just before transmission. If the origin computer is only unilevel secure, however, it contains no software trusted for assignment. The appropriate security level and computer name might therefore be appended by an adjoining multilvel computer (or by a certified correct, single-level computer) through which outgoing messages always pass. The local process and message names, however, can only be assigned in the origin computer since there is no way for an adjoining computer to know this information a priori.

Transmission

After the appropriate identifications are assigned to messages, they must arrive unaltered at their destination. If the logical link from origin to destination is only a communications line, encryption can be used to protect the identifications as well as the

message. But messages passing through switching computers must leave their identifications in clear text for switching control purposes. Therefore, both lines and computers on the logical link must be secure. Section V examines the security of these subsystems.

Authentication

While authentication of identifications has been analyzed for single-computer systems, networks require further consideration. They have more components to be authenticated and more alternatives for implementing the mechanism to perform the authentication. We examine these issues below.

Background: Authentication in Single Systems

In Reference 10, Burke distinguishes the internal environment from the external environment of a single computer. The internal environment is trusted benign since it is protected by physical, procedural and electronic boundaries. The external environment, on the other hand, is assumed to range from malicious and uncontrollable to partially benign. For a component in the internal environment to communicate with the external environment, a controlled interface must be established. In so doing, part of the external environment is temporarily brought into the benign internal environment.

The temporary extension is the responsibility of the I/O controls within the internal environment. Burke outlines three functions that the controls must perform for any sequence of input or output data transfers:

1) authentication;

2) controlled attachment; and

3) controlled operation.

Authentication confirms that the name of a component (which may be used to determine the maximum security level, the compartments, and the need-to-know attributes of the component) is in fact what it claims. Controlled attachment refers to the establishment of an I/O path and the transfer of control to the appropriate process handling the data transfer. Finally, controlled operation must ensure that the attachment is not changed and the security label is protected.

There are two groups of components in the external environment that may need to be temporarily connected:

1) terminals and I/O devices; and

2) users (on terminals) and media (on I/O devices; e.g., magnetic tape or printer paper).

Terminals and I/O devices are usually either physically secured in a host environment or else they are connected by an encrypted line to the internal environment. The physical protection or crypto handshaking provide authentication for such devices.

Users, and media for I/O devices, require software authentication mechanisms within the computer. The authentication procedure can be decomposed into two operations:

1) obtaining from the subject or object additional information (called authentication data) thought to be secret and available only from that subject or object; and

2) checking this with an expected result that is stored in a data base and indexed by each subject's or object's name.

Obtaining authentication data from users in a login procedure has been studied by Ira Cotton and Paul Meissner. They classify authentication data as being based on:

1) something the person knows;

2) something the person has; or

3) something the person is (11).

The first category includes the common passwords which are convenient but easily obtainable by watching users issue them, write them down, or tell them to another user. To be effective, they must be reissued frequently, or preferably, a one time pad approach should be used. A credit card with a magnetic strip is an example of something a person has, but credit cards may easily be mislaid, stolen, or copied. Signatures, hand geometry, fingerprints, and speech digitization are examples of supposedly unique characteristics, but automatic recognition of these characteristics is difficult and unreliable at present.

Obtaining authentication data for the media used on I/O devices is much less well defined. As Burke points out, a user can create the medium (e.g., a deck of cards for input or a new roll of paper for output) so there can exist no a priori list of authentication data. The solution may involve some kind of operator response to a query or some special operator procedures for manual authentication.

Mogilensky suggests, for example, that label spoofing for I/O devices connected to a single computer system can be avoided by reserving "some capability of the I/O device in question for use by certified software only" (12). For example, labels on line printer output could be surrounded by a special character which could be considered authentication data. If that character were missing when manually checked by an operator, he would know the label had been generated by uncertified software.

Authentication in Network Systems

A network system also has an internal and an external environment, both of which are very large. As in the single-computer system, all communication between the two environments requires authentication, controlled attachment, and controlled operation. We shall assume that these procedures are executed by each individual host in the internal environment when an external component attempts temporarily to attach itself to that host.

A major difference in networks is the addition to the external environment of dial-up hosts and the processes within them. Authentication of dial-up hosts, like terminals and I/O devices, could be performed by physical protection and/or hardware handshaking. SACDIN, for example, uses crypto, automatic calling, and modem handshaking procedures to authenticate a computer when an AUTODIN connection is first dialed. A DATE-TIME pair is also exchanged to prevent an intruder from playing a tape of the three handshaking procedures thereby falsely authenticating his computer as a host.

Authentication of processes is not necessary after host authentication if the hosts are trusted to provide and transmit identifications correctly as outlined above. If desired, however, a software mechanism to obtain and check authentication data for processes could be implemented.

Another major difference in networks is the potential for external components to connect to the internal environment in different places. Specifically, users may request access to host A

one day and a distant host B the second day. Traditionally, this is handled by distributing authentication data to each host a user may access.

Alternatively a network could be configured so that authentication data obtained at one host was checked at another. Then a user could logon any host (which permits him access) and that host would send his authentication data to be authenticated at the user's "home" computer or at a central computer where the information was stored.

Of course, centralization of an authentication data base would pose reliability problems if the central computer were to fail. But regionalization and a capability to specify alternative locations where a user is known would allow an authentication at other computers when his "home" computer was down.

AUDITING

Both access and authentication mechanisms result in an authorization to continue or a denial to receive service. A denial implies a security violation has been attempted and should therefore be recorded. Whether caused by a malicious intruder or a cleared user, the violation should be traced.

For these reasons auditing is important and the access as well as the authentication mechanisms should include auditing procedures. The auditing mechanism must be carefully designed, however, since a competent intruder will easily bypass it, particularly in an unverified system. Choosing a location for the auditing mechanisms is based on the location of the access and authentication checks themselves. To the extent that these checks are independently handled by the hosts, network auditing will resemble individual host auditing.

The records of access attempts should be sent to one or more of the network elements which have been assigned responsibility for security. (Assignment of security responsibility will be in accordance with the policy of network and/or host management.) A control center could provide efficient analyses of the limited number of violations expected, but sole reliance on a central element would weaken the capability for local management to act promptly. Also, the transmission of access records to the control center might themselves be liable to diversion, alteration, or destruction.

Determining what information is audited and sent to the chosen location requires careful consideration. If all access and authentication is recorded, the resource storing the data would quickly become flooded. Hence, a partial auditing - for example of violations only - may be desirable. Complete auditing, however, may be required by regulations and may be used to collect operational data as well as detected security violations. This data could be used to reconfigure the system more effectively or to perform statistical analysis. Alternatives for the use of auditing are being investigated by MITRE. (See Reference 13.)

SECTION V

SUBSYSTEM LEVEL ISSUES

Effective implementation of the network security model, and particularly of the network reference monitor, requires a secure communications subnet. Some of the capabilities needed for secure networking may be embodied in Network Front-End Processors, now under intensive development. Security issues relative to these and other subsystems are discussed in this section.

COMMUNICATIONS

Background

The communication subsystem consists of access circuits and the backbone or trunk circuits termed here the communications subnet. The security issues for access circuits (which are used only to transfer data to and from a single terminal) are essentially no different than for the circuits used to access a single computer installation. The approaches to protecting such circuits involve encryption of data on circuits which are accessible to persons who may misuse the data and physical protection of all terminals, encryption devices, and their interconnections.

The communications subnet transfers data between nodes. Since the number of nodes is usually too great to permit a fully-connected subnet, the circuits must be shared. Packet-switching techniques, as described, for example, by Cotton and Benoit (5), have been chosen because they permit efficient sharing of the high-speed circuits (50 - 230 kbps) which are needed to provide fast response to the user. Furthermore, many DoD networks (e.g., SACDIN, AUTODIN II, PWIN) are planning to use packet-switching technology.

Packet switching is accomplished by dividing long transmissions into packets of 1000 to 2000 bits. Each data packet is accompanied by 32 to 200 bits of control data identifying sender, receiver, security level, precedence, etc. Communications processors at each node of the network use the control data to forward the packet towards its destination. In addition, a Network Control Center or equivalent may be incorporated to monitor traffic flow through the subnet and to facilitate maintenance.

We will consider broadcast as well as switched distribution of packets. In broadcast systems such as the ALOHA network (15) or cable communications systems such as MITRIX (16), all packets are accessible by all nodes. Such systems are particularly useful when a substantial portion of the traffic is addressed to several users, where direct circuits would be inflexible or unavailable, or where the close proximity of the users permits use of shared, very-high-bandwidth circuits. They are likely to become more common as the size and cost of satellite ground stations and the cost of satellite circuits decrease, and as cable technology is developed for local networks.

Circuit Protection - Encryption Issues

It is obvious that encryption must be used to protect data on the communications circuits (since we have assumed that the network has a geographical extent broad enough to preclude the use of physical security for all circuits). It is not so obvious how cryptographic technology should be applied in a packet-switched network.

Link Encryption

Link encryption separately enciphers all data on each communications circuit. Its use requires cryptographic devices at both ends of each circuit, but these are shared by all traffic using that circuit. In most applications, the cryptographic keys are changed manually at regular intervals by a security officer. Some systems, however, permit remote keying.

Cryptographic protection is expensive. Major elements of cost are the cryptographic devices themselves and the vaults, guards, and other measures needed for physical protection. There are three approaches for lowering costs: reducing the cost of the cryptographic devices; reducing the need for physical protection; and sharing circuits (and hence their protection costs) between users.

Major reductions in the costs of cryptographic devices are being achieved through the use of LSI technology in large-volume production. The degree of physical protection needed is determined by the security level of the information.

Separate networks for classified and for unclassified traffic would be less expensive if the secure network required many fewer lines and cryptographic installations than would be needed for a single network. There are many reasons, however, to doubt whether a real saving could be realized.

1) Most nodes will handle classified data at least occasionally.

2) If the secure network were really smaller, it would have fewer redundant transmission paths and a lower availability.

3) There would be significant costs associated with operating separate secure and unclassified communications terminals.

The AUTODIN II specification includes explicit requirements for securing different processors to the degrees needed to handle different levels of sensitive information. If there were several different levels of security for processors, data of a given security level could only be switched by processors protected to that or a higher security level. The concept of different levels of switch security implies that different networks are embedded within the overall network. The embedded networks may not have enough nodes and circuits to provide the alternative routing and availability characteristics of the larger net. If not, this design implies a lower availability for transmission paths handling the most sensitive data than for transmission paths handling unclassified information.

We conclude, therefore, that a single, secure communications network will be most cost effective and that sensitive information (for which the network's security is considered inadequate) may be additionally protected. The ability to provide extra protection for some communications is discussed below (Mixed Systems of Encryption).

On the other hand, access circuits usually serve individual users and many will never carry classified data. It has been estimated that only 6% of the data terminal subscribers to AUTODIN service will handle classified data.

The major problem with link encryption is the presence of clear text data of every level of classification in the communications processors. We will discuss the resulting technical problems in the section on communications switch security.

End-to-End Encryption

An end-to-end encryption system enciphers data leaving the originating computer or user terminal and does not decipher until the data reaches the destination node or user. Clearly, this concept avoids the problem of having classified information in clear text in the switches, but raises a number of other problems.

1) Since many nodes (e.g., time sharing computers and terminal concentrators) support many simultaneous interactions, many cryptographic devices will be needed at each node.

2) Matching keys must be available to the widely separated pairs of crypto devices each time a connection is established. These crypto devices must be synchronized for each new connection.

3) The switching control information (e.g., destination address), which is an integral part of each message as it enters the communications subnet, must not be encrypted or the communications switches will be unable to function.

The need to install many crypto units at each node could be avoided, if multiplexed crypto devices were available. Such a device would store the current state of the crypto variable for each simultaneous interaction and apply the correct variable for the source or destination as each packet is received or prepared for transmission.

Some multiplexed crypto systems were developed in recent years using small computers to control arrays of key generators. Unfortunately, these systems failed to meet their cost objectives or were designed for switching nodes handling very large numbers of lines. An inexpensive multiplexed crypto device to handle a few lines is under development. The JTIDS cryptographic system is similar.

Key distribution techniques have been discussed by Branstad (18) and the System Development Corporation (9). Their concept is to have an "Agent" or "Security Controller" (a computer system in each case) responsible for distributing keys within a network or subnetwork. Whenever a process wishes to transmit to another process, it would request an appropriate key from the Security Controller. The request itself should be encrypted (to minimize spoofing or traffic analysis), using a key shared by the node and the Controller. The request being granted, the Controller would transmit keys to the originating and receiving processes. Their

hosts would then enter into direct communication, initiate synchronization, and begin to transfer data.

The Security Controller is not a practical concept at the packet level. There would be far too much overhead, too many keys required, and too long a response time for resynchronization. Therefore, keys should be distributed only for establishing connections. In effect, each host-to-host connection would obtain end-to-end encryption as it was initiated. Distribution of a large number of keys is still required, and a multiplexed crypto device is still needed.

Since control of the crypto keys must be a highly secure operation, the Security Controller might also be made responsible for granting or denying access rights (9, 18). That is, when a process requests a crypto key for communicating with another process, the Security Controller would test the relevant access privileges. The problems of implementing access control in a central security node are discussed in Section IV.

The mixing of control information and data in the packets creates perhaps the most significant problem in applying end-to-end encryption. Special encryption systems must be developed to separate control data from user information. An example of such a system is the Private Line Interface (PLI) developed by Bolt, Beranek, and Newman (8) under contract to DARPA. The PLI is expected to go into operational use in early 1976. This device combines an encryption unit with small minicomputers (currently the Lockheed SUE type). The software logic bypasses the crypto device during leader transmission, and subsequently enables it for text transfer. Since the functions performed by the minicomputers are not extensive, it is likely that they could be done by inexpensive microprocessors which might eventually be built into the crypto device itself.

Mixed Systems of Encryption

Combining link and end-to-end encryption may provide secure computer communications networks. NSA has done some work on a system called BLACKER (see Figure 9) which would combine these techniques. The circuit encryption is provided conventionally, but the end-to-end encryption incorporates certain access and interface control functions which are not yet defined.

Most likely these functions include the ability (as in the PLI) to pass communications control information around the encryption device. Also, provision would have to be made for generating keys and storing several crypto variables as for any multiplexed crypto

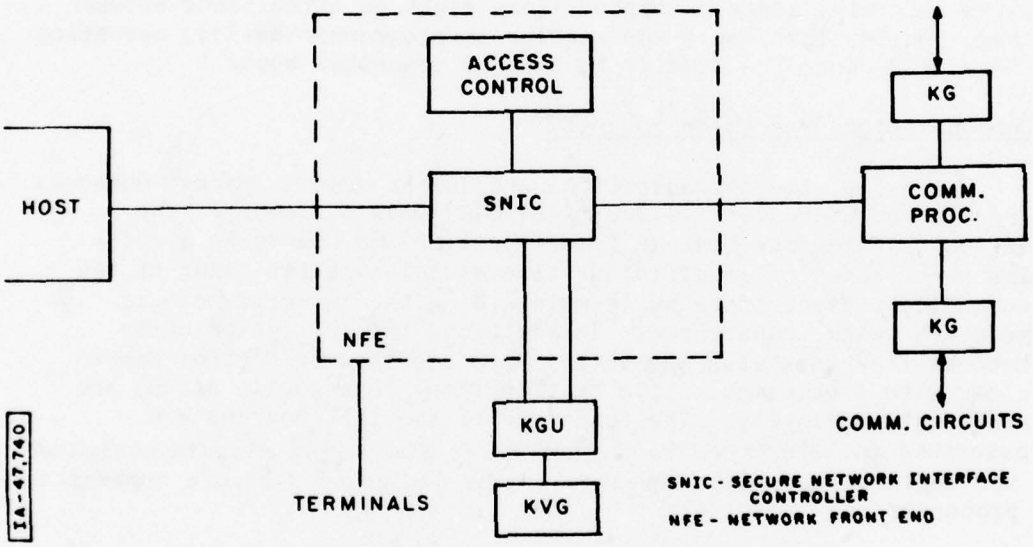


Figure 9. TYPICAL BLACKER CONFIGURATION

device. The use of both link and end-to-end encryption will significantly increase the cost of security. Two crypto devices per circuit will be needed plus a multiplexed crypto device per node (or perhaps for each processor at each node). One solution may be to restrict the use of end-to-end encryption to the highest levels of classification or to especially sensitive categories of information. Then only a few nodes would need the extra complexity of the multiplexed crypto devices.

These might be, for example, the nodes which in the AUTODIN II specification require extra physical protection. If both link and end-to-end encryption were used for messages to and from these nodes, the full reliability of the communications subnet would be available to the most sensitive data.

Unfortunately, these systems may be very expensive, if only low-volume production is required. If only a few hosts required the extra security, super-encrypted links might be established between them. These links could use regular cryptographic devices operating under NFEP control - similar to the PLI described above.

Communications Processor Security

Typically, the communications processors used in packet networks are minicomputers with 12,000 to 20,000 words of memory. The ARPANET's Interface Message Processor (IMP) may serve as a model. Its major function is efficient time-division multiplexing of the communications circuits while maintaining the integrity of the messages being transferred. In addition, the IMP periodically reports its own status and activity to the Network Control Center along with statistics on its traffic flow. Changes in status are reported immediately. The functions of the IMP programs are described in Refererce 19. An Air Force Study [14] also investigated the engineering issues involved in the design of a secure communications processor.

Within the communications processors, data of all security classifications handled by the system are multiplexed to and from the circuits. If only link encryption were used, the data in the processors would be in clear text and the processors would have to be designed to meet multilevel security requirements. Because the data is in clear text, the AUTODIN II specification imposes a misdelivery requirement on the communications switches of less than one packet in 10 billion. Perhaps, if end-to-end encryption were also used, less stringent security and misdelivery requirements would be imposed. We will assume that only link encryption is being employed so that we may examine the security implications for the processors and software. We will discuss first the security of the

switching functions of the processor and then explore how these processors may add to network security by performing some access checking functions.

A detailed study of the architectural requirements for a Secure Communications Processor (SCOMP) was carried out by Honeywell Information Systems (HIS) under contract to ESD/MCI (20). The applications for such a processor include use as an interface message processor or node in a computer network. The architecture proposed by HIS uses a Security Protection Module (SPM), interposed between the components of a standard computer, to perform the reference monitor functions (Figure 10).

The SPM is expected to require one or two large circuit boards and to degrade processor performance by less than twenty-five percent, possibly much less. In addition to implementing a reference monitor, it is important that the design of the communications processor software simplify the job of ensuring security. Two approaches may be useful in this respect: separating the classified information from the header information used by the communications processor; and implementing some access control

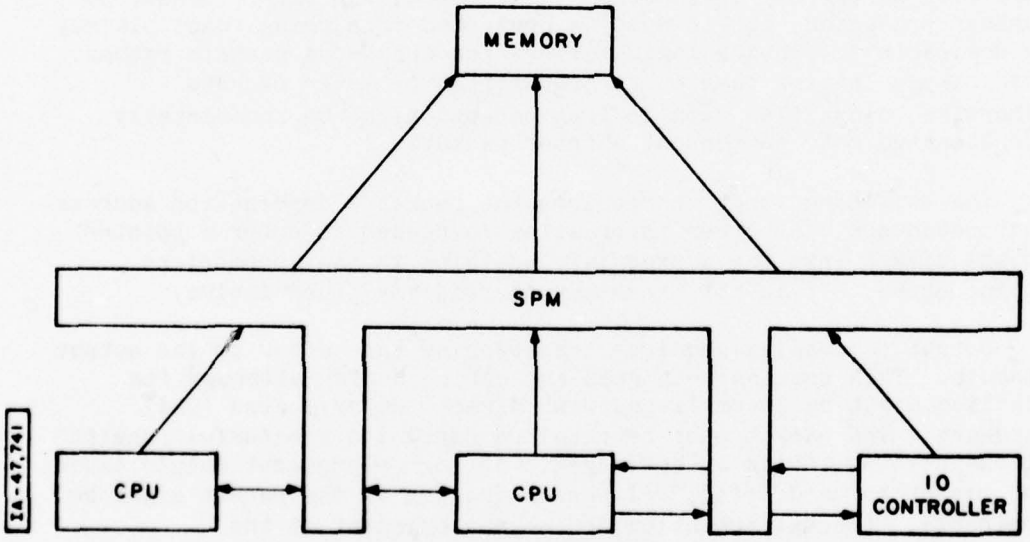


Figure 10 PROPOSED SECURITY PROTECTION MODULE (SPM)
(COURTESY OF REFERENCE 20)

functions to support the subnet view of reference monitor implementation.

Separating Control Data from Classified Information

Packets enter a communications processor memory from either trunk circuits or host access circuits (possibly from a network front end processor). The classification of the packet cannot be known until its header, containing this information, is processed. Hence, the buffers into which packets are entered, must be protected to the highest security level that the system can handle.

To determine the security requirements beyond this stage, we can examine the minimum essential functions that the communications processor should perform. These include:

a) error control;

b) switching; and

c) support for network control.

We assume that error detection on input will be performed by certified hardware. If an error is detected, the packet cannot be further processed, but it must be protected from being read. It may be desirable to provide logic for erasing erroneous packets rather than simply leaving them to be overwritten by later packets. Otherwise, classified data in long packets might be accidentally concatenated onto subsequent shorter packets.

The switching function requires the packet's destination address and precedence. No other information is needed to enter a pointer to the packet into the appropriate position in the appropriate output queue. It is not necessary to read any other fields.

Output processing requires transferring the buffer to the output circuit. This process must read the entire buffer although its function might be accomplished with direct memory access (DMA) hardware. The packet must be retained until its successful receipt at the next switch is acknowledged. An acknowledgement should cause the packet to be deleted; deliberate erasure of the packet might be desirable. (Packet retention and acknowledgement at the communications processor level would be eliminated if Cerf's analysis (21) were accepted. He argues that host-to-host acknowledgement and retransmission can provide improved network throughput.) The two-way transmissions required would cause a

security problem only when one host attempts to send a message of classification higher than it can receive to a second host only able to operate at the higher level.

Acknowledgement information for one packet may be included in the header of a packet going the other way. Hence, the acknowledgement process must be able to write the acknowledgement field of outbound packets.

In summary, if a multilevel secure switch were used, a thorough analysis might suggest that only a small portion of the software of the communications switch would need to be verified in detail. The following conditions may be sufficient.

a) The objects of major security concern (the packets) would be organized so that the text and unused header fields were separated from the header fields which are needed (and which are of low security classification). A logical separation is sufficient, but, if necessary, the text data could be stored in a completely separate memory, loaded and read by DMA techniques, but inaccessible to the communications switch software.

b) Only the precedence, destination, and acknowledgement fields need be read; the first two only by the switching process; the last only by the acknowledgement process. The ability to read the security label fields is also desirable as discussed in the next section.

c) Only the output process may read the entire buffer (this process will have to be verified; it might be implemented in DMA hardware).

d) Only the acknowledgement field of the header may be written and only by the acknowledgement process.

e) Otherwise, the buffer may be written only by the input hardware or erased only if:

1) erroneous data is input; or

2) the packet is acknowledged.

Of course, there are many software design implications in analyses of this type. Current IMP software apparently moves

packets within the IMP (22). The effect on performance of never moving packets would have to be investigated.

Security Checking Functions

The foregoing discussion has covered the communications processor's functions of forwarding packets. These processors also act as subnet entry points, receiving and transmitting traffic from the hosts (via a network front-end processor in some configurations). It may be desirable to require that access control functions be performed in support of the entry point tasks and perhaps in support of forwarding tasks depending on the level of implementation as discussed in Section III. While all processors in the subnet may be secured to the same high level, this may not be true of the hosts. Communications processors could check the security label of packets received from hosts against the security level of the host.

This simple check would help avoid overclassification of packets if packets having a higher classification label than the operating security level of the host are treated as erroneous. DoD 5200.1 requires scrupulous avoidance of overclassification. Furthermore, overclassification might violate the integrity of the subnet because an unsecured host might otherwise overload other hosts operating at higher security levels. (A similar kind of check of precedence level against authorized precedence usage should also be required.)

Also, to **prevent write down, a communications processor,** when transferring a packet to a host, should check the packet's classification and category against the operating security level and authorized categories of the receiving host.

These requirements, which are in the AUTODIN II specification (17), imply an additional operation on packet buffers - the reading of the security labels by the security checking function.

Control System

The processors in a packet-switching network exercise distributed control over the movement of packets through the system. They not only route packets through their own logic without

immediate central control, but they may also, in some designs, cooperate with neighboring processors to determine the appropriate interswitch routing. There do not appear to be significant security problems in the local flow control. The process(es) which performs this function needs information on the operation of the processor (e.g., queue lengths) and circuits (e.g., error counts) and does not need any information from the packets themselves. To the extent that such information is relevant to the health of the network, it might be interesting to an intruder, and should be protected.

The broader problems of system-wide flow control may, however, require a system-wide control mechanism. Furthermore, the detection, and more especially the diagnosis and correction, of failures is a complex and (with respect to a particular processor) infrequently required function. Hence, some designers provide a degree of centralized flow control (17, 22) and most use a Network Control Center (NCC) to support centralized management and maintenance.

The communications processors must support the NCC functions which include:

1) statistical analysis of traffic flows to detect bottlenecks;

2) detection and correction of circuit, equipment, and software failures;

3) correction of regional or system-wide flow control problems;

4) maintenance and updating of switch software; and

5) monitoring and control of the security system.

Clearly, the NCC has very extensive control over subnet operation. The extent of this control and the need to exercise it in real time imply the need for computer support in the NCC. The NCC function has important security implications for the design and operation of both the communications processors and the NCC itself. These implications may conveniently be analyzed in terms of the interactions between the communications processors and the NCC. There are two major types of interaction: those used for auditing

the operation of the system and those needed to modify the operation of the processors.

Auditing

Auditing interactions include those used for reporting traffic flows, errors, processor status, and security violations.

a) Traffic reports include such items as the number of packets entered, relayed, and output per unit time by precedence level and the length of buffer queues. The information is used by the NCC for monitoring long-term trends and helping to detect failures (which might be indicated by sudden increases in queue length).

b) Error reports include the number of CRC error checks detected for each circuit per unit time. They can provide early warning of a failing circuit.

c) Status reports provide general information needed for flow control.

d) Security violation reports are generated whenever a check of a packet's security label disagrees illegally with the authorized level for the host or circuit transmitting the packet.

The generation and transmission of auditing reports will not interact with the packet information and should, therefore, not directly threaten security. But it might be possible to transmit information at low data rates by generating patterns of, say, security violations (the confinement problem). The audit reports themselves may be sensitive since they describe the health of the network, but transmission through link-encrypted circuits should be adequate protection.

Control Actions

The NCC must be able to control directly each communications processor to:

a) set up test conditions (e.g., loop an interface on itself) and initiate test messages;

b) modify flow control parameters such as the precedence level permitted a particular subscriber;

c) initiate software traps to help analyze unusual status reports;

d) insert software patches to correct errors;

e) obtain copies of critical portions of software to check for errors or security violations; and

f) dump and load the entire program to correct errors and/or update the software.

Obviously, these control actions are a large potential threat to system security. The AUTODIN II specification recognizes this threat when it requires control actions to be initiated only under manual control and with supervisory approval. But NCC software will generate the actual action messages (and handle the responses) and, in the long term, some of these actions will become fully automatic.

Some of the control functions can be protected easily because they can be implemented by code normally resident in, and protected by, the communications processor, being executed only on direction from the NCC. In fact, the security kernel of the communications processor may be designed to protect all critical code from arbitrary modification, including itself. The use of the communications processor's kernel in this manner may limit the self-healing property of the comunications network since remote, on-line, reloading of the complete software may not be permissible. Manual reloading of a security kernel might be necessary. For example, SATIN IV allows only locally-initiated reloading of its Internal Access Control Mechanism. Alternatively, the critical software might be reloaded from a local Read Only Memory (requiring the periodic distribution of certified updates of the ROM to all communications processors).

The NCC computer is another vulnerable area, but one that can be addressed with kernel technology. The communication path to and from the NCC is more easily protected. The AUTODIN II specification implies that control actions will have, effectively, a very high classification. Effective protection of these messages may require the use of end-to-end encryption. The BLACKER concept (Figure 9) may provide such protection for hosts external to the communications subnet. The NCC will be such a host, but the communications processors are within the subnet. If BLACKER's

end-to-end encryption components were used where the circuits interface with the processors, they would garble normal traffic.

The solution may lie in logically separating the control processing from the switch processing and then inserting an end-to-end crypto device (which need not be multiplexed) between the two processing functions. There are no easy solutions here either and much work is needed.

Limited-Access Controls

So far, we have been considering the protection of the network from an external threat. In the multilevel network, however, some ostensibly legitimate users may not be entirely trustworthy and their right to use a terminal (even if only for unclassified functions) gives them access to the network and to its resources. Such a user may deny computer resources to other users or jam the communications subnet. His requests for service, even if denied at the processing nodes, can impede the free flow of legitimate traffic. The first limitation on his ability to jam the system lies in the bandwidth of the circuit available to him. It may be necessary to ensure that circuits which may be legitimately used by persons having relatively low security clearances have relatively low bandwidth. Such a constraint would not be too burdensome where simple terminals are employed, but as terminals using microprocessors and floppy disks (or other inexpensive memory) become increasingly common, their performance will be severely restricted by low-bandwidth access circuits.

A second possible limitation on the ability of an unclassified user to monopolize resources lies in dynamically restricting the volume of traffic at the entry point into the subnet. The communications processor which terminates the access circuit can be programmed to discard all or some fraction of the input from that circuit upon receipt of an appropriate control command. In a network, no one computer can be permitted to shut off service. It would be useful to devise a network control scheme which would implement an appropriate choke algorithm; such a requirement has been stated in the AUTODIN II specification.

Branstad (18) has discussed another approach to this problem: the use of a computer dedicated to access control to grant or deny permission for any user to access any resource. His discussion of the role of the "agency" indicates that it would:

a) test all requests for access to determine if they are legitimate;

b) deny improper requests; and

c) disable the terminal and, possibly, notify a security officer if the request(s) are "flagrantly" improper. (It is not clear how "flagrant" would be determined.)

Problems in the use of a centralized agency for controlling access are discussed in Section IV.

There are many other types of sabotage which can be suggested. Work currently underway on the integrity of computer systems will eventually have to be extended to include networks. Such an extension is beyond the scope of this paper.

Broadcast System Issues

Broadcast communications systems are increasingly being used and proposed for computer communications networks. The ALOHA system (15) has been in service for some years. The Joint Tactical Information Distribution System (JTIDS) is being developed. Experimental systems such as MITRIX (16) and the Distributed Computer System (6) are in active use. Because these systems are characterized by the fact that all data is generally available to all subscribers, they pose security problems somewhat different from those outlined above for switched systems.

JTIDS, though of most immediate interest to the Air Force, is a special case. Most subscribers need to process most broadcasts so that the system's navigation and command functions can be performed. Hence, JTIDS is not normally a multilevel secure system. This is not to say that the transmissions must not be encrypted - they still contain classified data - but all nodes must use a common key. Furthermore, since there are generally many nodes continuously active, frequent key changes will be needed.

Multilevel security within the data portion of JTIDS messages would be possible using end-to-end encryption external to the JTIDS processors. It is not known if a requirement exists for this capability, although the possibility has been discussed.

Other broadcast networks will be assumed to have multilevel security requirements. Although we are not aware of any planned use for ALOHA-type networks in the military, there is a strong trend for the costs of satellite circuits to go down much faster than the costs of conventional circuits. This trend will continue for some time so that broadcast data communications may often be preferred economically in the 1980s.

Broadcast networks, like the more conventional type, may have star or distributed topology. A star network, like ALOHA, has a base station communicating with many remote stations which do not communicate directly with each other. Only one large and powerful ground station is required. Distributed networks will have many, more or less co-equal, ground stations with the ability for any to broadcast to all, or at least to many others.

Star Networks

In a star network, since all stations communicate with the base station, the base station's processor will have to be a multilevel secure system handling the highest classification in the system. All classified transmissions must, of course, be encrypted. Furthermore, since we assume that the remote stations may operate at different security levels, the base station should use different crypto keys for communicating with each remote. Header and data fields of each message may be encrypted with the same key, since only the intended recipient will be able to decode the address. Since the remote stations are not expected to communicate directly with each other (although they may do so through the base station), they need not share common crypto keys. This approach, however, removes one operational advantage of broadcast communication: no longer can a single transmission serve all members of the network.

Where the broadcast capability is needed, special cryptographic transmission techniques will have to be used by, for example, sending a special control data stream to all remotes so that they will recognize that a subsequent transmission should be decrypted with a different key, one common to all. Significant overhead will be required to force recognition of the imminent special broadcast, but for a network of more than a few stations, the overhead is probably less than would be incurred in repeating the transmission with a different key for each remote station.

Distributed Networks

In a distributed system, the next packet may be broadcast by any of several nodes using any of several crypto keys, depending on the intended receiving node and process. There is no way for a receiver to know ahead of time which key to use for decrypting the message and no way to know if the message is even intended for this particular receiver until it has been decrypted. Of course, the headers could be transmitted in the clear, abandoning any vestige of transmission security.

There are several direct approaches to solving this problem. In one solution, all members of the network could use the same key for headers, and a different key (shared by the transmitter and the intended recipient) for data. This solution retains the ability to broadcast a single, encrypted message to all participants without having to rekey all the crypto devices, but may involve excessive overhead for key distribution.

A second approach might involve the use by all receivers of simultaneous multiple keys. To determine what key to use, the initial portion of every transmission could be decrypted using every key simultaneously. Whichever keystream yielded the recipient's address would continue to be used for subsequent decryption.

Transmission security cannot be fully achieved with the type of broadcast system currently used - time-division interleaving of complete messages. Not only is every transmitter readily located, but the volume of its output is readily measured. In a system using fixed packet slot assignments (e.g., JTIDS), the same is true, but significant improvements can be made by interleaving at the bit level because the brief duration (one bit time) of the signal from a particular transmitter will restrict the enemy to a low probability of direction finding.

Cable Networks

Cable communications systems have some characteristics similar to those of radio broadcast systems. The communications paths of the cable systems are, however, strictly limited and hence may often be physically protected. Link encryption is generally not practical because the data often uses only one frequency band on the cable while other bands are used for other types of information (e.g., video). Hence, in most installations physical protection will be preferred. Link encryption may be useful when two cable systems are

to be joined (for data transmission purposes) by a different kind of circuit.

End-to-end encryption is usable for cable systems. As with the satellite broadcast systems, the headers will be susceptible to traffic analysis and Trojan Horse threats (unless a common key is used for all headers). However, it isn't clear that the engineering has been done to make end-to-end encryption practical for cable systems. Very high performance will be needed from the crypto devices and it may be necessary carefully to synchronize all (perhaps several hundred) such devices in a sizable net.

Summary

From the discussion above, we can identify four important issues that particularly affect the communications subnet.

1) The Network Control Center is critical in packet-switching networks because it will likely have the capability for modifying the communications processor software. Not only is a multilevel secure computer system going to be needed in the NCC, but significant additional security steps should be used. For example, the personnel must be cleared to the highest security level appearing in the network and the system must be designed to ensure that supervisory approvals are obtained for software modifications.

2) While simple link encryption is necessary, it may not be sufficient for particularly sensitive data. A careful tradeoff analysis is needed to balance the cost and performance penalties of using additional end-to-end encryption (for at least some classes of data) against the cost (in the absence of such encryption) of additional physical security and the risk of hardware or software failure at the communications switches. Additional study is also needed of the cost and performance tradeoffs of securing circuits and switches to different degrees appropriate to different levels of data classification.

3) Key distribution, in both packet-switched and broadcast communications systems, will be an important issue if end-to-end encryption is used to supplement link encryption. There appears to be a strong requirement for multiplexed cryptographic technology to support end-to-end encryption in a computer network, and to reduce the costs of encrypting access circuits. The design of the key distribution system will have both performance and security implications.

4) It is obvious that the communications processors must be designed to meet multilevel security requirements. It seems likely that careful analysis and design can minimize the software functions that must be trusted; such an analysis should be carried out. The possible effects of hardware failure must be examined. The communications processors can also perform some access control functions, especially in identifying the source of messages entering the communications subnet so that source labels cannot be faked (without subverting the communications switch).

USER STATIONS

The security of the data terminals with which users access the network is important to network security, but the issues are no different than for terminals accessing a simple ADP system. Current practice relies primarily on physical security of the terminals, which are usually enclosed in expensive vaults. Development of a secure terminal which can be used in an ordinary office environment has started; in this section we summarize this project. Additional discussion on some points is included.

Secure Office Terminal Program

The objective of the Secure Office Terminal program is to develop a Secure Communications Controller (SCC) incorporating a cryptographic device for use in the SOT. When not in use, the SCC (or at least its crypto device) will be locked in a safe; when needed, it will be plugged into a terminal designed to meet TEMPEST requirements. In a somewhat different configuration, the SCC will be used as a multiplexed device able simultaneously to encrypt and decrypt several terminal circuits where they are interfaced at a front end processor.

Secure Office Terminal - Issues

Storage

Volatile storage technology may be used for buffering data in the terminal, but the design features that ensure erasure of the memory when the terminal is switched off will have to be certified. Although unattended operation of the terminal would be useful for some message handling applications, the need to provide a safe or vault for the incoming messages would negate the basic benefits of

the SOT concept. Message storage, if required, should be provided in the communications processor.

Error Handling

Error detection (and correction by retransmission) will be used, but some designs of crypto devices may permit a single bit error to contaminate several successive characters. More complex retransmission protocols than are normally used with data terminals would be required to handle this phenomenon of error extension. A buffer for blocks of data and the ability to ensure that each block is correct before outputting it would be valuable.

Operation of the Cryptographic Device

If a Security Officer were required to key each crypto device each time it was to be used, there would be a heavy demand for Security Officers. Remote keying is highly desirable. Multiple key storage might be an effective alternative, if the key changes can be done by the terminal operator or remotely.

Unclassified Use

The ability to use the terminal for unclassified access to the network without the crypto device would be valuable. It should be usable by either cleared or uncleared personnel.

Terminal and Controller Identification

Unique identification for each terminal and controller is desirable, since some access rights should only be exercised from a few, specially protected, terminals. But the identification system must be protected from tampering, if the restriction of special functions to specially protected terminals is to be an effective extension of the system's security. Critical parts of the terminal which might be bugged or bypassed should be protected by alarm circuits. An alarm system which is itself secure while used in an unprotected office environment may not be readily achievable.

User Authentication

Secure terminals now in use are physically protected so that the authentication of the user's identity is often implicit in his having access to the terminal. His password is an inexpensive extension of this authentication. When SOTs are in unsecured office areas, the only formal user authentication will be the password,

which is probably insufficiently secure. Some authentication is implicit, however, in the user's access to the safe to get the SCC. More complex user authentication techniques may be required; some possibilities are discussed in Section IV.

Smart Terminals

There is an increasing trend for incorporating processing power in data terminals. To the extent that the processor and its program may be tampered with, there is an additional threat to the system's security. For instance, classified data may be left accidentally in the processor's memory or small programs added maliciously to the intended code may hide classified data in auxiliary memory or transmit it to another network node. Almost certainly the user should be denied programming capability in any terminal support processor. There will be too many such processors scattered all over to permit certification of their code. Firmware programs protected by alarm systems may be necessary. An alternative concept would be to permit program loading only from the Network Control Center after terminal logon. Then the NCC might be responsible for ensuring the security and integrity of the terminal support processor's code.

Summary

The major security problem posed by data terminals will arise from the incorporation of storage and processing capability. If expensive vaults for such terminals are to be avoided, certifiable techniques will have to be used for controlling the memory and software.

Key distribution is another important problem in this area, but one that is likely to be solvable through remote keying.

To the extent that unique and unalterable terminal identification is considered necessary, secure means for preventing tampering must be devised.

NETWORK FRONT-END PROCESSORS

The use of a network front-end processor (NFEP) to handle communications and terminal support functions for a large computer is becoming increasingly common for network hosts. Network front-ends have been recommended for the Prototype WWMCCS Network and for the AFSCNET. The functions of these NFEPs include:

a) host-to-host protocol implementation;

b) host-to-communications processor interface; and often

c) terminal support.

The first two of these functions, commonly included in the Network Control Program (NCP) of a host, should be performed in an NFEP to:

1) simplify change and development of the host-to-host protocols, e.g., to encompass the changes suggested by Cerf (21);

2) reduce the costs of implementing NCPs for new hosts;

3) reduce the interference with host functions and increase throughput; and

4) minimize the number of different, divergent NCP implementations.

An NFEP serving a secure network host must, of course, be secure. Security requirements for communications processors were studied by Honeywell Information Systems, (HIS), under Contract F19628-74-C-0205 as part of the ESD/MCI Security Program (23). Honeywell's arch tectural analysis suggests use of a Security Protection Module (SPM) added to a standard commercially-available computer to implement the reference monitor functions. Their report includes extensive discussion of design and performance issues (20).

Since the security requirements for an NFEP are little different from those of a communications processor, the concepts and design proposed by HIS may be satisfactory for networking. In fact, a secure communications processor of the type they describe might also serve as the processor in the communications subnet. The major function that might be required of an NFEP, and which has not been studied by HIS, is support of multiplexed encryption. For instance, if the secure communications processor were to serve as the Secure Network Interface Computer in the BLACKER concept (Figure 9), additional security and architectural constraints might be imposed. The potential effects on design and performance of handling this additional function need to be investigated.

SECURITY OFFICERS

Assuring the security of a network will require an appropriate allocation of responsibilities among the managers of the network, on the various hosts, and of the communications subnet. Unless a host is dedicated to network service, its local management will not be able to surrender responsibility for its security. The connection to the network, however, may introduce security problems that local management is powerless to solve alone. The proper allocation of security responsibility must be addressed in a practical application; no general solution appears practicable.

Whatever the allocation, the security officers will have important jobs. If the system design does not facilitate their work, the operational constraints on the users will become burdensome and the system security and/or utility will suffer. The security officers' major duties will be to:

a) maintain the access control and user authentication mechanisms;

b) distribute cryptographic keys;

c) supervise the functioning of the Network Control Center; and

d) monitor the operation of the system.

The issues in each of these areas have been discussed in this paper. Clearly, the security officers must have access to all the most critical portions of the network and must have computer aid for their work. The Network Control Center might seem the logical base for exercising security officer functions, but it must be remembered that the NCC is not usually a critical component of the net. Most network functions will not be affected, in the short term, by failure of the NCC. On the other hand, the security officer functions, particularly key distribution, are essential. Therefore, at least some of the security officer functions should probably be decentralized.

SECTION VI

SUMMARY AND RECOMMENTATIONS

MAJOR NETWORK SECURITY ISSUES

As a result of the considerations presented in this report, we conclude that there are six particularly important issues in computer network security. These are issues which are unique to networks, or of much greater importance for networks than for independent computer systems.

Distribution of Control Mechanisms

The network components which control access to resources (including data), which control security and distribute cryptographic keys, and which control the operation of the communications subnet may be centralized or distributed and may be fully or partially integrated. The optimum degrees of integration of these control functions, and of centralization (if any) are not obvious. The tight control and global viewpoint available through centralized operation must be balanced against the unreliability and lack of flexibility.

Distributed access control may be implemented at host, host-communications network interface, or in the communications network. A combination of all levels is likely to be most effective in maintaining security in a network which includes both multilevel and unilevel computers of different security levels.

Identification and Authentication

Because of the need to transfer authentic user and process identification (including security level) between distant computers, special care must be taken to ensure that the identification information attached to every transmission is generated correctly and transmitted without modification. Appropriate naming conventions, certified I/O software, and secure communications are needed to achieve these ends. Message labeling, like access control, will need to be distributed between multilevel hosts and the communications subnet (or multilevel NFEPs) acting on behalf of systems which are not multilevel secure.

Encryption Techniques

Link encryption must be used for the communication subnet's main circuits, and for some access circuits. End-to-end encryption should also be used at host interfaces where particularly sensitive data may enter the network. A mixed encryption system can provide both security and the high transmission path availability promised by networks having multiple transmission paths. Key distribution problems for the end-to-end encryption need to be solved.

Communications Processors

The communications switches must be designed for multilevel security. A promising start has been made on the basic architecture in the HIS study (20), but additional work is needed. The quantity of certified software necessary may be minimized with careful design.

Network Front-End Processors

Although the Secure Front-End Processor development program may solve most of the security problems for Network Front-End Processors, end-to-end multiplexed encryption techniques need to be integrated into the designs.

Network Useability

The requirements of network security will impose some burden on the network's users, in addition to password entry. The need for a user authentication mechanism, the performance limitations imposed by the key distribution and access control subsystems, and the security officer's work may seriously inconvenience users, if the system is not carefully designed.

RECOMMENDATIONS

Efforts should be continued to resolve the major security issues identified above. Since the first phase of the AUTODIN II system is to be developed soon, especially early attention to the communications processor design issue is important. A government effort to parallel the expected contractor efforts would be highly desirable. In this connection, the analyses by Cerf (21) of the performance of the ARPANET IMPs should be taken into account.

At the same time, work should be started on developing the cost, performance, and security vulnerability tradeoffs involved in alternative implementations for the various control mechanisms.

It is expected that work on alternative encryption approaches will be continued by NSA without additional stimulus.

The need to ensure the correctness of identifications transferred between network components should be embodied in design specifications which define and allocate the functions that must be performed. The specifications should ensure the integrity of these functions which are necessary if a network access control mechanism is to function effectively.

REFERENCES

1. R. R. Schell and P. A. Karger, Security in ADP Teleprocessing Systems - ADP Host Role, Electronics and Aerospace Convention, Washington, D. C., Oct. 1974.

2. J. P. Anderson, Computer Security Technology Planning Study, ESD-TR-73-51, James P. Anderson and Company, Fort Washington, Pa., October 1972.

3. D. E. Bell and L. J. LaPadula, Secure Computer System: Unified Exposition and Multics Interpretation, ESD-TR-75-306, Electronic Systems Division, AFSC, Hanscom AFB, MA, March 1976 (ADA023588).

4. A. J. Neumann, A Guide to Networking Terminology, N. B. S. Technical Note 803, National Bureau of Standards, Washington, D. C., March 1974.

5. I. W. Cotton and J. W. Benoit, Prospects for the Standardization of Packet-Switched Networks, Proceedings of the Fourth Data Communications Symposium, Quebec City, Canada, October 1975.

6. D. J. Farber and F. R. Heinrich, "The Structure of a Distributed Computer System - The Distributed File System," Proceeding of the 1st International Conference on Computer Communications, Washington, D. C., October 24-26, 1972. Available from IEEE Computer Society, 8949 Reseda Boulevard, Northridge, California.

7. V. G. Cerf and R. E. Kahn, A Protocol for Packet Network Intercommunication, IEEE Transactions on Communications, Vol. 22, No. 5, May 1974.

8. Specifications for the Interconnection of a Host and an IMP, Report #1822, Appendix H, Bolt, Beranek, and Newman, Inc., Cambridge, Mass., January 1976.

9. K. Auerbach, An Analysis of WWMCCS ADP Security, Vol. II - Data Communication Networks, TM-WD-5733, System Development Co., 31 March 1974.

10. E. L. Burke, Concept of Operation for Handling I/O in a Secure Computer at the Air Force Data Services Center (AFDSC), ESD-TR-74-113, Electronic Systems Division, AFSC, Hanscom AFB, MA, April 1974 (AD780529).

11. I. W. Cotton and P. Meissner, Approaches to Controlling Personal Access to Computer Terminals, Proceedings of the 1975 Symposium: "Computer Networks: Trends and Applications", National Bureau of Standards, Gaithersburg, Maryland, June 18, 1975.

12. J. Mogilensky, A General Security Marking Policy for Classified Computer Input/Output Material, ESD-TR-75-89, Electronic Systems Division, AFSC, Hanscom AFB, MA, September 1975 (ADA016467).

13. C. Engleman, Audit and Surveillance of Multilevel Computing Systems, ESD-TR-76-369, Electronic Systems Division, AFSC, Hanscom AFB, MA, April 1977 (ADA039060).

14. The Feasibility of a Secure Communications Executive for a Communications System, MCI-75-10, Electronic Systems Division (AFSC), Hanscom AFB, MA, August 1974.

15. N. Abramson, "The ALOHA System," Computer - Communications Networks, Franklin F. Kuo, ed., Prentice-Hall, Englewood Cliffs, New Jersey, 1973.

16. D. G. Willard, "A Time Division Multiple Access System for Digital Communication," Computer Design, June 1974.

17. System Performance Specification for AUTODIN II Phase 1, Defense Communications Agency, November 1975.

18. D. K. Branstad, Security Aspects of Computer Networks, AIAA Conference on Computer Network Systems, April 1973.

19. The Interface Message Processor Program, Bolt, Beranek, and Newman, February 1973, AD-A-008-876.

20. "Analysis of Secure Communications Processor Architecture," Honeywell Information Systems, Inc., November 1975.

21. V. G. Cerf, An Assessment of ARPANET Protocols, Proceedings of the Jerusalem Conference on Information Technology.

22. A. A. McKenzie, The ARPA Network Control Center, Proceedings of the Fourth Data Communications Symposium, Quebec City, Canada, October 1975.

23. ESD 1974 Computer Security Developments Summary (Tasks 11-17), MCI 75-1, Information Systems Technology, Applications Office, ESD/MCI, December 1974.