AD-A043 446 ARMY COMMAND AND GENERAL STAFF COLL FORT LEAVENWORTH KANS F/G 9/2
COMPUTER SECURITY FOR ASSIST.(U)
JUN 77 M H BEACH

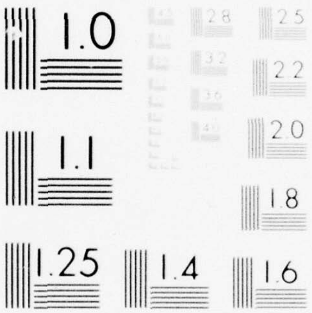
UNCLASSIFIED

NL

| OF |

AD
A043446

END
DATE
FILMED
9-77
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

AD A043446

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

2

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
6 Computer Security For ASSIST.

5. TYPE OF REPORT & PERIOD COVERED
10 June 1977

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s) 10
~~Major~~ Martin H./Beach

8. CONTRACT OR GRANT NUMBER(s)
9 Masters thesis

9. PERFORMING ORGANIZATION NAME AND ADDRESS
CGSC
Fort Leavenworth, Kansas 66027

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
US Army Command and General Staff College,
Fort Leavenworth, Kansas 66027

12. REPORT DATE
11 10 June 1977

13. NUMBER OF PAGES
12 58p.

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
No restrictions; distribution unlimited.

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

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

DDC
RECEIVED
AUG 15 1977
RECEIVED
B

18. SUPPLEMENTARY NOTES

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

AD NO.
DDC FILE COPY

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

This thesis examines the multilevel security problem of simultaneous processing of compartmented and collateral data at the Intelligence Data Handling Site, Forces Command Intelligence Group, Fort Bragg, North Carolina. Existing security controls are examined, and a list of software controls are discussed to reduce the risk of penetration, whether accidental or deliberate. Software controls are described in four major areas: access controls, input/output controls, residual controls, and audit → over

037260

20. contd.
trail controls. The security kernel is discussed as the heart of all software controls. A method of verifying the software is discussed and a procedure is explained for certifying the ASSIST system as possessing an acceptable security risk. Recommendations are described to reduce the risk of penetration and certify the system as secure through software controls.

COMPUTER SECURITY FOR ASSIST

A thesis presented to the Faculty of the U. S. Army Command and General Staff College in partial fulfillment of the requirements for the degree

MASTER OF MILITARY ART AND SCIENCE

by

MARTIN H. BEACH, MAJ, USA
BBA, University of Georgia, 1971

Fort Leavenworth, Kansas
1977

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

MASTER OF MILITARY ART AND SCIENCE

THESIS APPROVAL PAGE

Name of candidate Martin H. Beach

Title of thesis Computer Security for ASSIST

Approved by:

_________________, Research Advisor

Matula_________________, Member, Graduate Faculty

_________________, Member, Consulting Faculty

Accepted this 3rd day of June, 1977, by

_________________, Director,

Master of Military Art and Science.

The opinions and conclusions expressed herein are those of the individual student author and do not necessarily represent the views of either the U. S. Army Command and General Staff College or any other governmental agency. (References to this study should include the foregoing statement.)

ACCESSION for
NTIS White Section ☑
DDC Buff Section □
UNANNOUNCED □
JUSTIFICATION Refer on Page 47
BY Removed AP.
DISTRIBUTION/AVAILABILITY CODES
Dist. AVAIL. and/or SPECIAL
A

ABSTRACT

This thesis examines the multilevel security problem of simultaneous processing of compartmented and collateral data at the Intelligence Data Handling Site, Forces Command Intelligence Group, Fort Bragg, North Carolina. Existing security controls are examined, and a list of software controls are discussed to reduce the risk of penetration, whether accidental or deliberate. Software controls are described in four major areas: access controls, input/output controls, residual controls, and audit trail controls. The security kernel is discussed as the heart of all software controls. A method of verifying the software is discussed and a procedure is explained for certifying the ASSIST system as possessing an acceptable security risk. Recommendations are described to reduce the risk of penetration and certify the system as secure through software controls.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Tables Page

CHAPTER I

INTRODUCTION

Problem Definition

There is presently no capability to process compartmented and collateral intelligence simultaneously within the Army Standard System for Intelligence Support Terminal (ASSIST) computer. The problem is penetration, accidental or deliberate, of compartmented intelligence data by users who are not granted the appropriate level of access. Current security control measures physically disconnect the user from the system at the intelligence data handling site. The only method for processing compartmented intelligerce is a segregated mode cf operation. For the purpose of this thesis, the Forces Command Intelligence Group (FORSIG) site located at Fort Bragg, North Carolina, will be examined.

Secure System

A computer system is secure if it is known to prevent all actions defined as unauthorized by security specifications. Penetration studies conducted by the Department of Defense (DoD), involving several different systems, have demonstrated that existing shared, general purpose systems are not secure. In all such systems, a malicious user can construct a program that can defeat the access constraints supposedly enforced by the system. To be secure, all possible ways to perform unauthorized actions must be blocked. No way to circumvent the protection mechanism can exist.

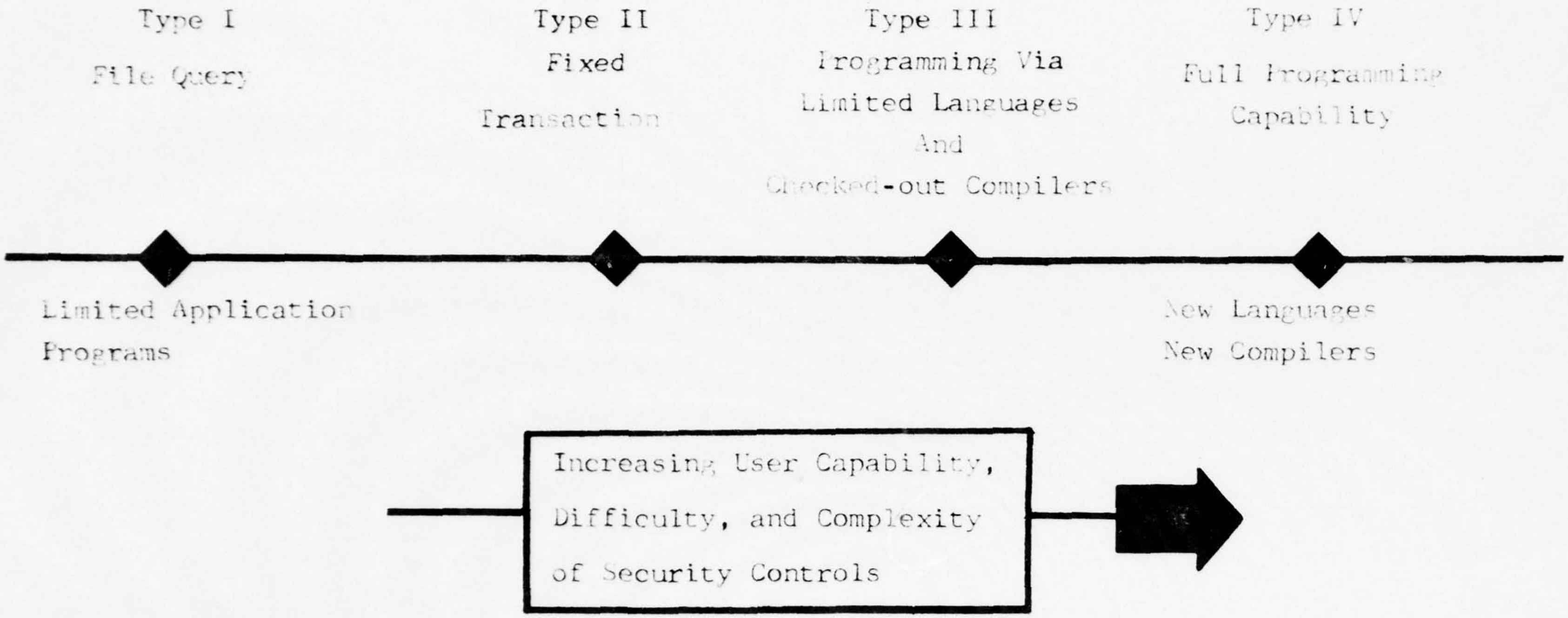
Computer Security is an all-encompassing term which includes physical aspects, personnel, administration, hardware, communication, and software. Traditionally, the method of securing the computer has been to remove the entire system to a protected environment. The compartmented data within the computer has been afforded the same protection as non-computerized data. The ASSIST resource-sharing computer systems, whereby the computer capabilities and components are shared by many users or many jobs, have compounded the security problem of safeguarding compartmented data. Resource-sharing allows a number of users to interact within the computer while giving each user a variety of options depending on the capabilities. The more user capabilities offered by the computer, the more difficult and complex are the security controls. Figure 1 is a graphical description of this situation.

This figure illustrates that users with limited programming capabilities within a system do not pose as serious a security problem as users with unlimited capabilities. The system itself, by the type of accesses and processes offered, increases the difficulty and complexity of the security controls.

The ASSIST system is a remote-access system incorporating the capabilities of the file query and fixed transaction systems shown in Figure 1. Therefore, capabilities offered by ASSIST do not require the most complex security controls, as would a system offering greater programming capabilities, such as the TYPE III and IV systems shown in Figure 1.

File query user capabilities of ASSIST enable intelligence analysts at the intelligence data handling sites to execute only limited application programs. The analyst does not have the capability to alter the program, although the capability exists to couple several of these

User Capabilities



Type I
File Query

Type II
Fixed
Transaction

Type III
Programming Via
Limited Languages
And
Checked-out Compilers

Type IV
Full Programming
Capability

Limited Application
Programs

New Languages
New Compilers

Increasing User Capability,
Difficulty, and Complexity
of Security Controls

Figure 1[1]

programs together and insert parameters into the selected programs.

The ASSIST's fixed transaction capability allows the intelligence analysts to insert parameters. Programming is limited to input language symbols provided by the ASSIST monitor software. The symbols are not used to construct an internal machine languzge program that can subsequently be executed upon command from the user. Thus, the user does not have the opportunity to obtain control of the computer directly, because he is buffered from it by the interpretive software.

Multilevel Security

Different levels of data accesses are processed at the intelligence data handling site. Since not all users have the same level of access, a multilevel security problem is created. A multilevel security mode of operation provides the capability of various levels of classifications and compartments of data to be concurrently stored and processed in the automatic data processing (ADP) system. In the ASSIST remote access system, the data can be selectively accessed and manipulated from terminals controlled by various personnel having different security clearances and access approvals.[2] So the problem associated with multilevel security at Forces Command Intelligence Group (FORSIG) is controlling those users who do not have the appropriate security clearance for access to the system when compartmented material is being processed. The multilevel security problem may not be completely solved, but security controls can be designed to bring the risk to an acceptable level.

The term acceptable risk level in a multilevel ADP operation is not formally defined by any authority. The ADP Security Manual discusses implementation of a secure resource-sharing ADP system which processes classified data so that with reasonable dependability, accidental or deliberate penetration can be prevented.[3] The term reasonable dependability indicates that a certain amount of risk can be tolerated. The Defense Science Board Task Force on Computer Security stated the following concerning multilevel utilization:

> Since a complete proof of protection is not within the present state of the art,. . . it is recommended that the system designer estimate the probability of occurrence of a single failure or the combination of failures that could result in a disclosure of classified information. Based on this information, the Responsible Authority can determine whether the risk probability is acceptable or not.[4]

The Defense Science Board included "special caveat information" (compartmented intelligence) in its discussion on multilevel utilization.[5] The Defense Intelligence Agency Manual states that "computer systems require multiple security measures and procedures to attain an acceptable level of security."[6]

The ADP Security Manual, the Defense Science Board, and the Defense Intelligence Agency Manual on Security of Compartmented Computer Operations all indicate that there is, at some point, an acceptable risk level. However, the Department of Defense authorities have not yet defined controls necessary for an acceptable risk level in a multilevel ADP operation.

Purpose

The purpose of this thesis is to define those controls necessary in a final design that will bring the risk of penetration to an acceptable level. The design will be determined by examining the current state-of-the-art of security control measures and determining their application in solving the multilevel security problem at the ASSIST intelligence data handling sites. Through this examination of current state-of-the-art control measures and existing security control measures in force at the intelligence data handling sites, a system of controls will be described that, when implemented, will reduce the risk of penetration. A design of software security controls for the IDHS FORSIG, Fort Bragg, North Carolina, will be the main emphasis of this thesis.

Background

ASSIST was designed to give the intelligence analysts a system that provides ready access to all data in intelligence files related to their needs. ASSIST supports the linking of Army intelligence data handling sites and allows communications with other Department of Defense (DoD) systems. Figure 2 depicts ASSIST sites throughout the world where analysts can interact with local or remote intelligence files to exchange data with other analysts.

Figure 3 shows how the intelligence analysts interface with Department of Defense systems and other ASSIST sites. The World Wide Military Command and Control System (WWMCCS) computer, a dual Honeywell 6060, is located in the Pentagon and is used for ASSIST host-support services (remote job entry and time-share access). Communication links from the intelligence data handling sites (IDHS) and the Assistant Chief of Staff for Intelligence (ACSI) provide tne interface to WWMCCS. A switch at the office of the ACSI permits IDHS access to Defense Intelligence Agency (DIA), and then to the Defense Intelligerce Agency On-Line Intelligence System (DIAOLS) and the Community On-Line Intelligence System (COINS).

The ASSIST system offers a powerful, user-oriented ADP base for intelligence data handling system's analysts. The system also offers a means of integrating the strategic and tactical intelligence analyst's problem by having ASSIST terminals at the tactical field location of the

ASSIST Sites

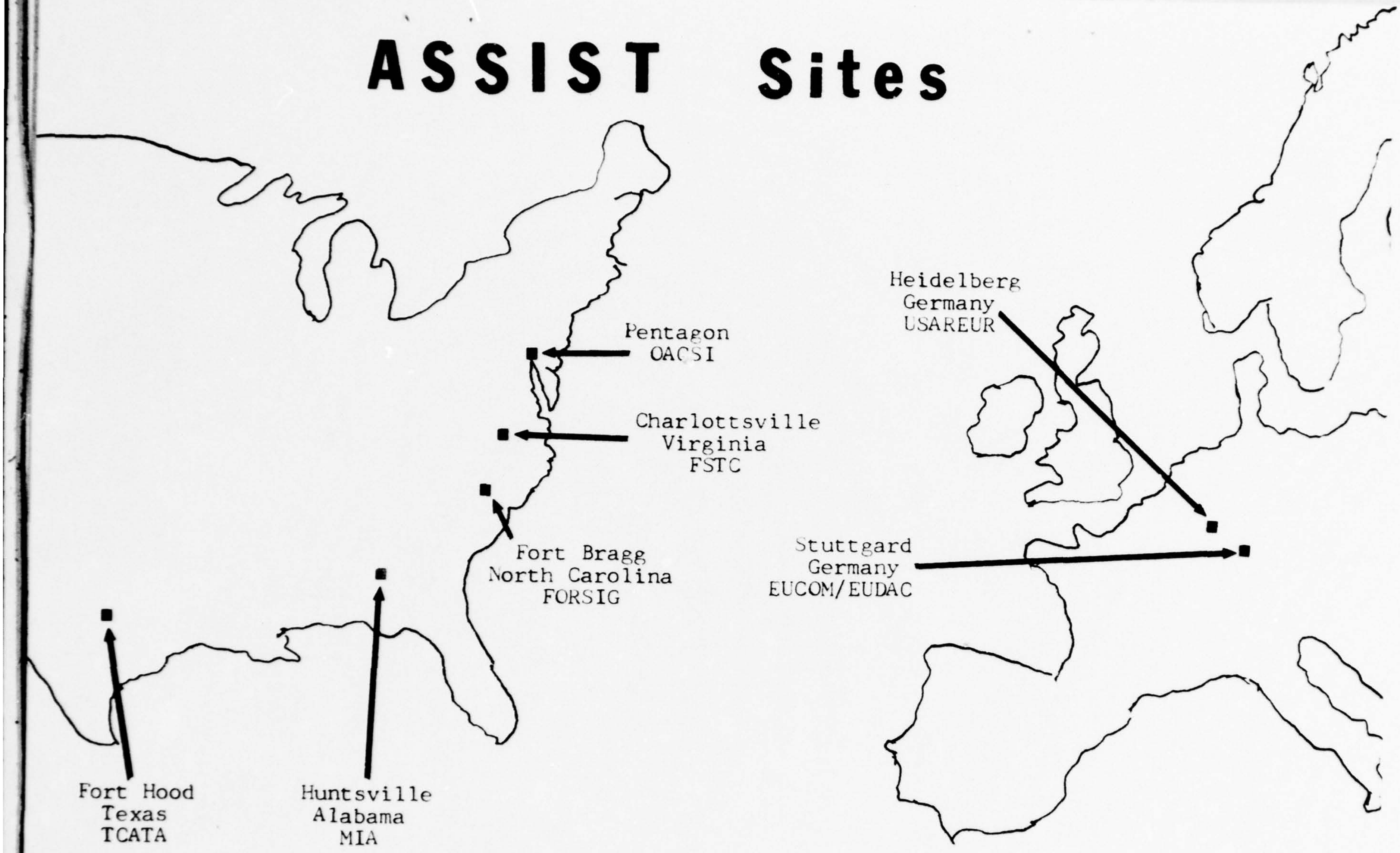
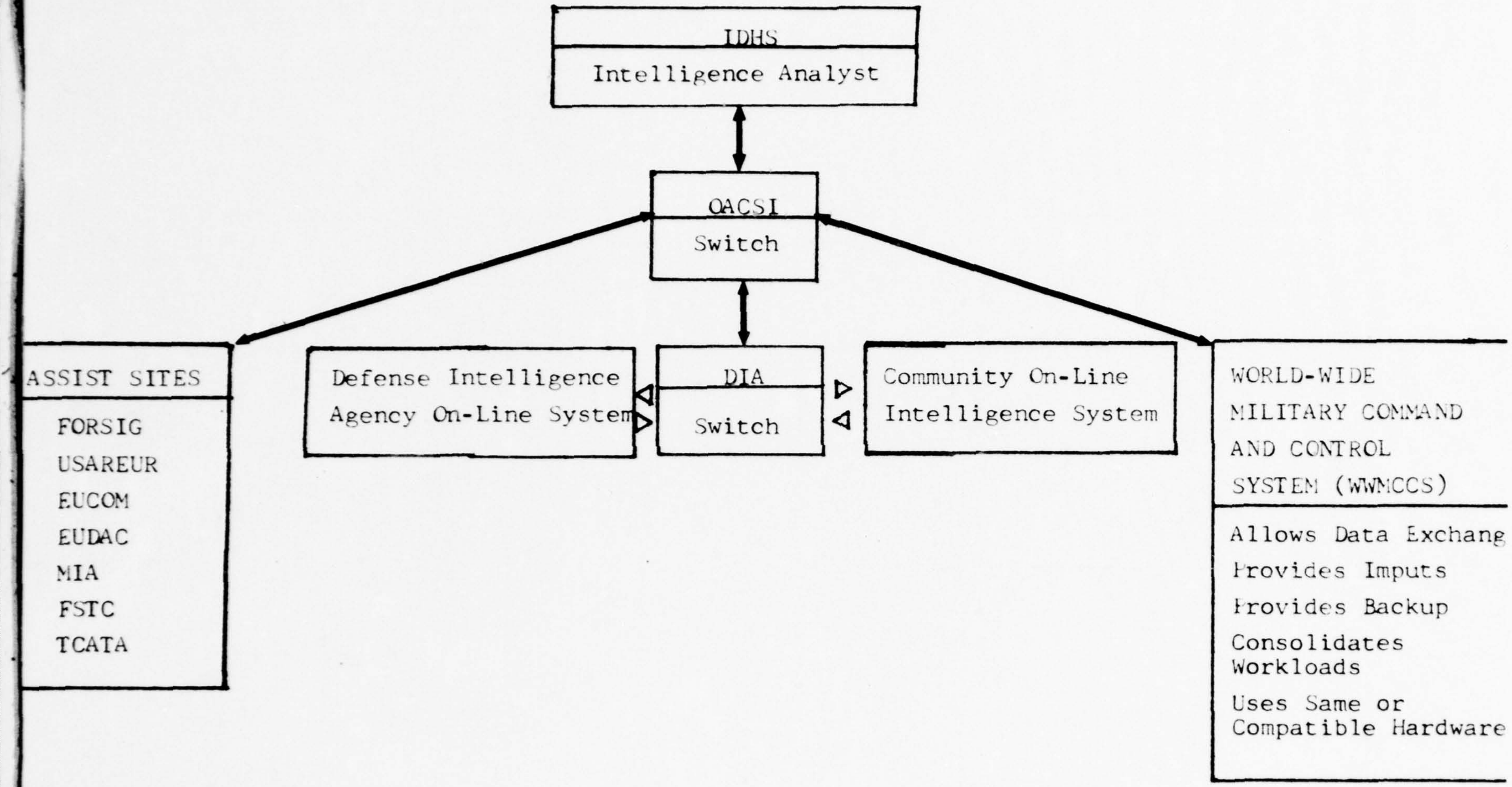


Figure 27

ASSIST Capabilities



IDHS

Intelligence Analyst

OACSI

Switch

ASSIST SITES

- FORSIG
- USAREUR
- EUCOM
- EUDAC
- MIA
- FSTC
- TCATA

Defense Intelligence Agency On-Line System

DIA

Switch

Community On-Line Intelligence System

WORLD-WIDE MILITARY COMMAND AND CONTROL SYSTEM (WWMCCS)

- Allows Data Exchang
- Provides Imputs
- Provides Backup
- Consolidates Workloads
- Uses Same or Compatible Hardware

Figure 3[8]

maneuver corps headquarters.

ASSIST Security Requirements

ASSIST multilevel security requirements can be categorized as physical, personnel, communication, hardware, and software. Access control must be in force throughout the system. Physical, personnel, and communication access controls have been identified in current DoD regulations, and will not be discussed in detail as they apply to IDHS.[9] The IDHS FORSIG meets these requirements.

Access to the computer from the terminals has to be controlled by the software of the system. Physical, personnel, and communication security controls are currently accepted by DIA as secure in the handling of compartmented and collateral material, concurrently in a manual system. These same controls are also applied in an ADP system. Software controls in an automated system can replace the Special Security Officer who is used to insure security in a manual system. The necessary software security controls have not been identified by any DoD authority.

Software Security

Software security controls insure that the security constraints placed on the system are enforced.

> The systems designer contributes to security by capitalizing on the facilities of the computing systems in order to augment the external manual procedure. Specifically, he can design and program more elaborate, more precise, and more consistent controls over selective access to sensitive data. These controls, coupled with personnel, procedural and physical measures taken by data-processing-operations management can significantly reduce an organization's exposure to potential data security problems.[10]

Software control measures can be grouped into four areas: access, input/output, residual, and audit trail. Each control measure is a separate area or control point which works together with the others to reduce the risk of penetration.

The emphasis in all multilevel security studies is the recognition of the user and his authorization level. "Once a user is identified, the system must determine what he is authorized to do. He may be authorized to use some programs or functions, but not all."[11] Terminal identity is of equal importance to user identity at the intelligence data handling site, for it is the location of the terminal that governs its authorization level.

When the user/terminal is properly identified, security flags (authorization code) estabiished by software will accompany each request for data. The security flags will determine what data the user and/or terminal are authorized input/output. The user and/or terminal identification/location controls the input/output authorization.

After the user has concluded his transaction or query, main memory or peripheral devices contain residual data not intended for use outside the context of the process. This residual data is a potential security hazard unless it is erased to prevent unauthorized users' access.

The software technique used to verify that the system is operating correctly is the audit trail, a system of logs that record how, what, when, and where a user interacts with the system.

The design of the software controls will be examined in Chapter II. Chapter III will examine IDHS at FORSIG and how the designed controls reduce the risk of penetration.

CHAPTER II

EXAMINATION OF RELATED LITERATURE AND DISCUSSION

Introduction

This chapter examines state-of-the-art software security controls. The literature that was examined discussed security requirements in a multilevel system. None of the literature dealt with compartmented intelligence, although the systems did deal with varying levels of classification. Compartmented intelligence "includes only that intelligence material having special controls indicating restrictive handling for which systems of compartmentation or handling are formally established."[12] The term classification refers to the three levels defined by DoD as top secret, secret, and confidential. Compartmented intelligence is classified data with special caveats and a need to know. Collateral intelligence is all other intelligence classified without special caveats.

Multilevel systems deal with various levels of authorization determined by levels of access given individuals or terminals. This thesis treats compartmented data as just another level of access, which is governed by special access privileges to those individuals and/or facilities meeting stated DoD criteria.[13]

Software security design is developed from an examination of existing studies devoted to multilevel security. The software security control design, as proposed in the following paragraph, are for the FORSIC IDHS facility and may or may not have application to other systems or ASSIST sites.

ASSIST Software Security Control Designs

The choice of applicable software security controls for the ASSIST intelligence data handling sites is based upon the selection of techniques that provide proper protection of compartmented data from uncleared users. The Defense Science Board Task Force on Computer Security, in discussing automation of a multilevel security system, described the following operating environmental features:

> Integrity for both itself and the security system; multiprogramming or on-line, interactive capability; a basic file system; protection (read, write, and execute) for users from each other; a secure method of identifying and authenticating users; an interface with the security system that permits input/output for any user only after authorization by the security system.[14]

The following is a list of controls that have been identified as system's design requirements to reduce the risk of penetration.[15] Each category of controls will be discussed in a later section.

Access Controls

1. Identify correctly the terminal in relationship to the transaction and/or query authorization.
2. Identify correctly the user in relationship to the transaction and/or query authorization.
3. Recognize repeated attempts to gain entry.
4. Prevent changes or modifications in fixed transaction key words.
5. Insure user ability to request transactions or queries.
6. Recognize repeated errors in transactions or queries.
7. Protect the user's process from interfering with other user results.
8. Restrict access to files resulting from queries by the originator or other identified user.
9. Disconnect (log out) terminals by means other than physical

disconnection after a specific period of inactivity.

10. Establish the access rights of both the user and terminal.
11. Deny the request for data at a higher level of access than authorized for user or terminal.
12. Recognize requested data's classification in transactions or queries.

Input/Output Controls

1. Determine originator and location of a transaction or query request.
2. Determine classification of the file resulting from a query.
3. Control the release of files resulting from queries.
4. Identify specific terminal addresses.
5. Control routing of information.
6. Display the classification of hard-copy or screen output.
7. Assign highest level of classification to a composite data request.
8. Assign classification to individual data elements.

Residual Control

Obscure classified data after process is complete.

Audit Trail

1. Alert control group if illegal terminal ertries are made or attempted.
2. Trace entire messages to originating terminal.
3. Monitor the extent of activity at remote terminals.
4. Determine and log all requests for compartmented data.
5. Insure constant internal checks to insure correct software and hardware functions.

The characteristics listed above were extracted from existing systems and from research on proposed systems. Further, literature related to multilevel security and fixed transaction/free form query systems with remote terminals that process mixed classification data was also examined.[16] It is recognized that implementation of software security techniques may be very expensive, but all are considered necessary to reduce the risk of penetration.

Discussion of ASSIST Software Control Designs

The discussion of software controls describes preventive action the software will encompass. Software controls prevent users, without the proper level of access, from gaining access to compartmented data.

Access Controls. The location of the terminal and its proper identification is one of the most important controls. The identification of the terminal location dictates the level of access authorized. This identification should be automatic, which is possible because of direct communication lines between the terminal and the computer. This automatic identification, accomplished when the terminal is turned on, would verify the terminal authorization as established in the terminal profile table. As a double check on the system, a manual identification should also be used. The identification would be accomplished by the user entering a predetermined code with the log-on procedure.

The user access authorization is accomplished as the user logs-on by use of a password, which is compared to a profile table listing the level of authorized access. The password also indicates the file functions (read, write, or execute) allowed the user. When this ability to recognize errors in passwords or processes identifies possible unauthorized access attempts, the terminal should be automatically disconnected from the system. The system security officer (SSO) should then be notified and be allowed to determine the reason for the error before allowing the terminal's reconnection. The disconnection process would be a software lockout requiring the SSO to exercise privileged instructions for release. Correct instructions to each user will assist in preventing false alarms.

The protection of user's process and access to files from other users is controlled by bounds controls. Each user is assigned memory space by software and each memory reference is tested to be sure it falls within the bounds. "Memory space is further protected by the user's inability to generate addresses that are outside their own assigned memory space."[17]

The profiles and lists are the stated access authorization levels to all entries and processes in the computer. The following profiles and lists are needed to insure segregation of compartmented data from users not authorized the stated access level:

1. User and terminal access privilege.
2. File access list (who is authorized entry).
3. File access profile (what function user is authorized).
4. Data element profile (classification level).

All access control profiles and lists must be afforded tamper-proof protection to preserve the integrity of the system.

Input/Output Controls (I/O). "As data is entering and leaving the system, the security control information (classification and categories) associated with it must be transmitted as well as the data itself; and for some I/O devices there may be a maximum data classification and categories."[18]

The input/output controls described above could be implemented easily with hardware. The controls are not implemented on the PDP 11/45 computer which is located at FORSIG. Thus, the "reference monitor," which validates all input/output authorizations, must be implemented with software.[19] The system notifies the user of his classification level and prevents a terminal from receiving or sending data higher than authorized. For each transaction or query, the software checks the access rights in the terminal and user profile tables before performing the operation. After the operation is completed, the data classification/category is checked against the terminal and user profile tables to insure authorization level before any output occurs.

Residual Controls. All magnetic recording devices retain an electromagnetic image of the recorded data after the initial impression. Since both primary and secondary storage in most on-line multiuser systems are used repeatedly, it is possible that an area could have stored compartmented data that could be assigned to a user not authorized access. To prevent this from happening, the software overwrites memory space after it is deallocated. Properly functioning I/O controls would not allow this data to exit the system to a terminal not authorized access. All controls function together to form a secure system. A method to insure that all these controls function properly is the audit trail.

Audit Trail. The audit trail is a series of logs that record all transactions and queries within the system. The audit trail is an after-the-fact review by the SSO to determine what actions have transpired which affect the security operation of the system. Table 1 is a list of automatic logs and their contents.

Another method of proving the security of a system is a security verification program. This program provides a continuous check on the security of the system's operations. Actual responses are compared to

TABLE 1[20]

1. SYSTEM ACCESS LOG
   - a. MODE OF ENTRY (RJE, TIME SHARING, OR BATCH)
   - b. IDENTIFICATION OF TERMINAL
   - c. IDENTIFICATION OF USER
   - d. TIME/DATE BLOCK
   - e. TIME USED
   - f. I/O DEVICE DEDICATION
2. FILE USAGE LOG PROTECTED
   - a. RECORD OF OPEN FILE AND CLOSE FILE
   - b. IDENTIFICATION OF USER ACCESSING FILE
   - c. ACTIVITY TAKEN AGAINST FILE (READ, WRITE, MODIFY, EXECUTE, ETC.)
   - d. IDENTIFICATION OF TERMINAL ACCESSING FILE
3. SUSPECTED VIOLATIONS LOG
   - a. TYPE OF SUSPECTED VIOLATION
   - b. IDENTIFICATION OF TERMINAL
   - c. IDENTIFICATION OF USER
   - d. ACTION TAKEN*
   - e. DATE/TIME BLOCK
4. TRANSMISSION LOG
   - a. IDENTIFICATION OF TERMINAL RECEIVING/ACKNOWLEDGEMENT
   - b. IDENTIFICATION OF USER REQUEST
   - c. IDENTIFICATION OF FILES INVOLVED
   - d. DATE/TIME BLOCK
   - e. IDENTIFICATION OF COMMUNICATION PORT/LINE
5. SECONDARY STORAGE LOG
   - a. RECORDED AREA OF MEMORY ASSIGNMENT BY CLASSIFICATION
   - b. TIME AREA DEDICATED
   - c. TIME AREA RELEASED

*Original thought.

known correct responses to verify that the system is performing properly. This type of program needs to be updated periodically to continually test the system.

To insure security, software controls have to be implemented in a way to make them tamper proof. The most appropriate vehicle for tamper-proof implementation is the security kernel.

Security Kernel

A security kernel is a protected core of software whose correct operation is sufficient to guarantee enforcement of constraints on access, and is the basis for a secure system. "All protection mechanisms are collected in the kernel, so that only this kernel need be considered in order to verify that the specified security properties are implemented correctly."[21]

> A characterization of the mechanism that should be included in a security kernel can be obtained by viewing the security specification as a set of constraints on the interaction of the various computations that occur in a computer system. The protection mechanisms of the system prevent one computation from exerting an unauthorized influence on the input, progress, or output of another. Permanently stored data is one form of the input and output of computations.[22]

By viewing the entire security requirements of the system, it can be said that three principles must exist with the security kernel. First, the kernel must be tamper proof; second, it must always be invoked; and third, it must be small enough to be subject to analysis and testing to assure correctness.[23]

The tamper-proof condition is essential. If the kernel's software can be altered either by programming or manually, its integrity cannot be guaranteed; thus, there is no security certification.[24]

The continuous invocation of the kernel in all accesses, fixed transactions, queries, and input/output, is the heart of the security feature. This can be accomplished by a reference monitor validator which validates all references (to programs, data, files, input/output, etc.) made by programs in execution against those authorized for the user and/ or remote terminal. The reference monitor validator not only assures that the references are authorized to share resources, but also that the reference is the proper kind (i.e. read/write/execute). If the kernel is not invoked on each transaction, then again, security cannot be guaranteed.[25]

Finally, the condition that the kernel must be small enough to logically demonstrate that it is complete, faithful to the security designs, and correctly implemented, is another way of saying it must be capable of enforcing stated security constraints on access to information in the system. Being proven correct must be a continuous procedure.

One method currently in use by the Defense Intelligence Agency is an on-line security monitor that performs not only software checks, but also hardware checks, to insure security-related controls are working. "The program size of tne monotor is 2,000 words. 110,000 hardware checks and 10,000 software checks are performed during each shift."[27] This type of check helps to detect software and hardware failure, so deficiencies can be corrected.

The security kernel design incorporates the reference monitor validator, access control (to the system), and authorization mechanism. Further, it will probably incorporate the administrative programs to represent and maintain user and program authorizations. The kernel is visualized as the center for software security controls. The Electronic System Division Computer Security Panel identified the concept of a reference monitor and security kernel as fundamental to a secure computer system.

The kernel should also allow verification through formal techniques. The kernel will enforce access constraints that combine the controls reflecting the information-release policies of the military security system and the controls on information sharing within the ASSIST system. The broad constraint on the ASSIST system is to allow multi-users to process and not allow compartmented intelligence data to be transmitted to users not granted the appropriate access. The kernel's functions consist of:

a. Identifying and authenticating each user/terminal requesting to access and process classified and/or compartmented data.

b. Insuring that the communication of security authorization of user and terminal is transmitted with each transaction or query request.

c. Insuring that compartmented intelligence output is transmitted only to those devices authorized to receive compartmented intelligence.

d. Monitoring and requesting job termination when abnormal conditions arise.

To accomplish its functions, the kernel must have a segregated, controlled area within central memory that controls access to tne kernel's software and makes it tamper proof.

The Multics Corporation has developed a prototype security kernel and implemented it on the Digital Equipment PDP 11/45 hardware. The PDP 11/45 is the heart of the intelligence data handling site at FORSIG.

In order to be efficient in a general-purpose system, it does require hardware support. You could do it with software, but it

> would not be efficient. And again the primary hardware characteristic we find necessary for the efficient implementation of the kernel is the segmented virtual memory with independent access rights per segment and at least three machine states or security domains. In the case of the PDP 11/45, instead of just having a master and a slave, they have a kernel mode, a supervisor mode, and a user mode. It is the existence of those three states that allows us to implement the kernel mode.[28]

Summary

This chapter has examined software controls needed at the FORSIG intelligence data handling site. It also described the security kernel and the important role it plays in reducing the risk of penetration to an acceptable level. The penetration risk associated with the IDHS at FORSIG and how software controls counter that risk are examined in Chapter III.

CHAPTER III

ASSESSMENT OF THE RISK AT THE IDHS FORSIG

Introduction

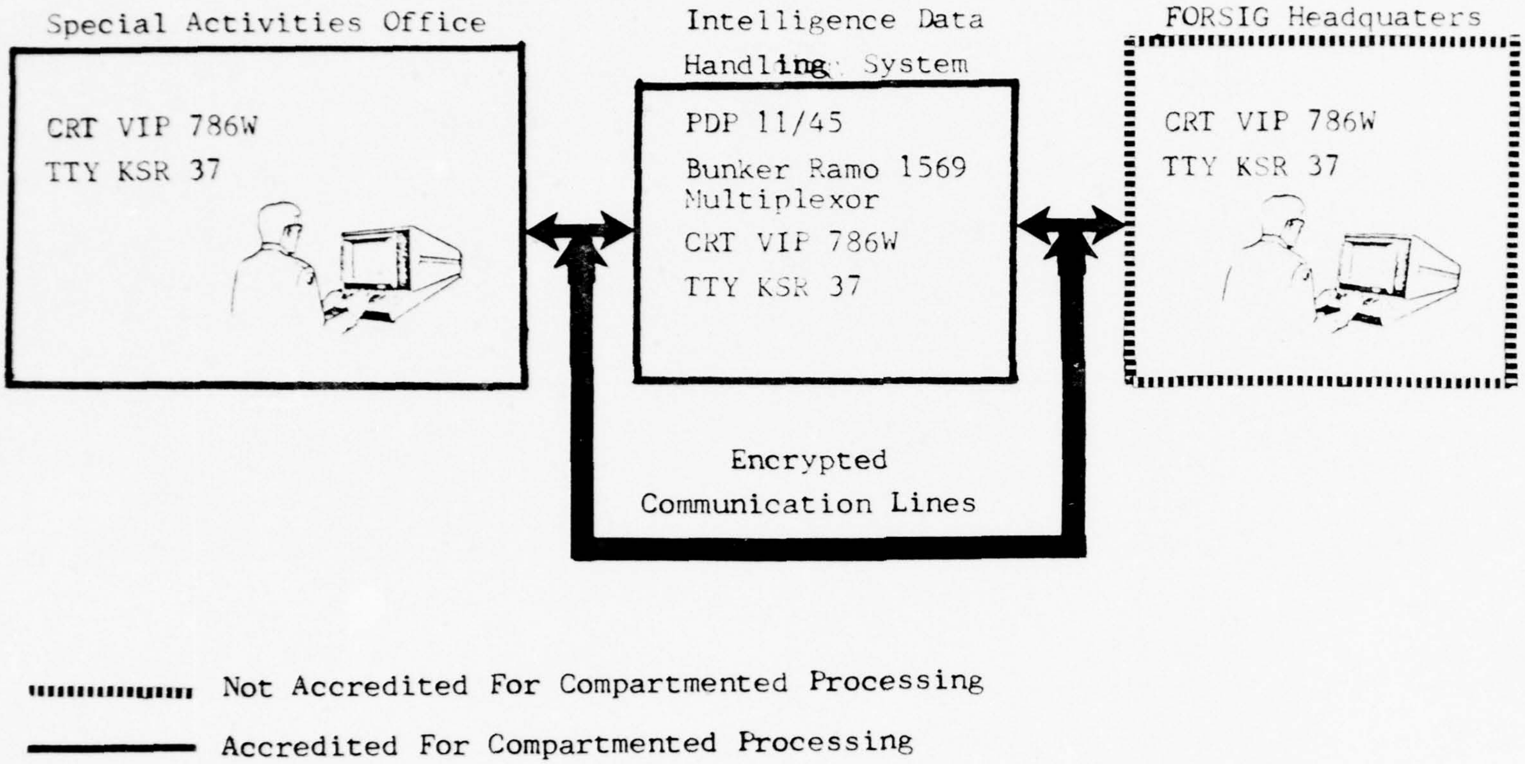
Risk assessment is a difficult task. The only logical method is to identify all possible weaknesses in a system and to design a control to counteract that weakness. The environment surrounding the ASSIST intelligence data handling site FORSIG must be examined to determine its weak points. This chapter examines the environment, the penetration risk, and the method controls either in force or in design to reduce the risk of penetration.

IDHS Environment

The ASSIST system at FORSIG has three separate facilities, all connected to each other, which form the intelligence handling system. Figure 4 depicts the ASSIST configuration at FORSIG. Terminals are located at three separate facilities: the intelligence data handling site, the special activities office, and the FORSIG headquarters. The equipment available to the analysts is the cathode-ray tube (CRT) and teletype (TTY) input/output devices. The CRT does not produce a hard copy and is used primarily to input data and update the onsite files and other processes not requiring a hard copy. The TTY is used when the analyst does desire a hard copy. These I/O devices allow the analysts to accomplish the following:

1. Manage local files.
2. Transmit messages, inter-intra sites.

The ASSIST IDHS At FORSIG



Special Activities Office

CRT VIP 786W
TTY KSR 37

Intelligence Data Handling System

PDP 11/45
Bunker Ramo 1569 Multiplexor
CRT VIP 786W
TTY KSR 37

FORSIG Headquaters

CRT VIP 786W
TTY KSR 37

Encrypted Communication Lines

▨ Not Accredited For Compartmented Processing

— Accredited For Compartmented Processing

Figure 4[29]

3. Invoke operation of application programs at the IDHS.

4. Interact with OACSI host computer for remote job entries and time sharing.

5. Transmit files between ASSIST sites.

6. Transmit and receive data between the analyst and DIA.

All three facilities are connected by encrypted lines to the computer located in the intelligence data handling system site.

The facilities of the special activities office and the intelligence data handling site are accredited to process and store compartmented intelligence. The FORSIG headquarters terminals are located in an adjacent vault constructed to physical specifications to allow processing and storing of compartmented intelligence standards, but is not accredited to do so. Accrediting the facility and using it for compartmented operations would eliminate those analysts not appropriately cleared to receive such data.

The FORSIG headquarters, which is not accredited to receive compartmented intelligence, is the location that does the most processing. It is also this site where the problem of multilevel security exists, since the other sites are cleared for the compartmented level.

This multilevel security problem is currently being solved by processing data in a 'system high mode', whereby all the data are controlled in accordance with the highest classification or category level being processed in the system at the time. There are manual provisions to raise or lower the security level and also extract material at a lower classification. Users located at the FORSIG headquarters are physically disconnected from the system during the time compartmented intelligence is being processed. The set procedures utilized for raising and lowering the system operating level are those listed by the Defense Science Task Force on Computer Security for a segregated mode of operation. This disconnection creates a problem, however, since approximately 75 percent of the analysts are disconnected from the system for periods that result in the loss of both time and manpower.

Currently, compartmented data is being processed two days of the week. Because files are not being updated in a timely manner, this is an unsatisfactory practice. As other ASSIST sites become operational, a heavier demand will be placed on the system to process compartmented data.

There are two possible solutions. One is to upgrade the FORSIG headquarters vault to the compartmented intelligence level. Such action would mean that analysts who are not cleared for compartmented intelligence would be denied access to the terminals and to that particular facility. It would also increase the work load of the special security officer who must screen all material leaving his control to insure that no compartmented intelligence material is compromised. All personnel working in the FORSIG headquarters have a minimum of a secret clearance and approximately twenty (20) percent have a compartmented intelligence clearance. Because all analysts are not clearable, and the clearance procedure normally requires six months after the initiation of paperwork, facility upgrading would limit available analysts' time and manpower.

The second possible solution would be to implement software security controls that would allow both the collateral and compartmented intelligence data to be processed at the same time. When implemented, this would eliminate loss of time and manpower and make the computer available to all the analysts regardless of clearance.

Contrasting the two solutions, the first would appear to be better because of ease in implementation. Of the 75 percent of the analysts located in the FORSIG headquarters, only 20 percent are appropriately cleared to process compartmented intelligence, causing a disadvantage, by losing approximately 55 percent of the total analysts available for processing. The current method eliminates the uncleared personnel two of the five working days, or 40 percent of the time. Personnel turnover, lack of qualification by some users to qualify for access tc compartmented intelligence, and lengthy clearance procedures make upgrading the facility unsatisfactory.

The FORSIG intelligence data-handling system's operations can be summarized by the following:

a. System's use--multilevel (fixed transaction and free form query).

b. User environment--physical protection of terminals, controlled access to terminals, and protection of communication lines with cryptographic techniques.

c. Threat--deliberate and accidental penetration.

d. Authorization--collateral and compartmented.

IDHS Penetration Analysis

Penetration of the ASSIST system would be by extraction of data from the facility or the system by unauthorized persons. Penetration can be deliberate or accidental. Deliberate penetration occurs when a person, not authorized access to restricted data, makes an attempt to receive it. Accidental penetration occurs when the system, for various reasons, fails in the security constraints it is designed to enforce.

Deliberate Penetration. Due to secure communication lines and physical protection afforded remote terminals, it would be highly improbable for a deliberate penetration to be accomplished by an unauthorized user. "The installation of secure communication links for all terminals on the system effectively prevents any external penetration attack, and forces a penetration agency to seek an alternate method."[30]

The secure communication links protect only from an external source. An attack could be mounted from within, from an authorized terminal. The physical protection, alarm systems, and the access control to the terminals significantly reduce the risk of an unauthorized user gaining access to the terminal.

The degree of threat posed by an authorized user is directly related to the number of security controls he must bypass or render inoperative, and once inside, the amount of programming he can do. The ASSIST system at the intelligence data handling sites is a fixed transaction/free form query system as discussed in Chapter I. The user capability affecting the operation of the system is limited by the intrinsic capability of the tools he can use. Properly operating software and hardware within ASSIST do not provide the user with sufficient tools to take control of the system. The intruder cannot attack the system with his own program.

The user may be able to gain unauthorized access to compartmented data by probing the system for weak points caused by errors or logic oversights. Trap doors are created by support personnel that allow circumvention of security techniques in the programming and operating systems supporting the application. The security threat posed by this type of operation depends on whether the application is designed in such a way as to assure that each user is fully controlled in all actions he may take on the system. It is therefore, imperative that both the application programs and operating systems for the supporting hardware be implemented by appropriately cleared personnel. This action should prevent the possible inclusion of trap doors.

Because the ASSIST system has been implemented by appropriately cleared personnel, the probability of a deliberate threat from outside the system is considered small. The fact that all users have at least a secret clearance reduces the probability of a deliberate threat from within, though not eliminating the possibility.

Contrasted to deliberate penetration, accidental penetration is a problem. The next section will discuss this possibility and the controls suggested to counter it, as well as an inside deliberate threat.

Accidental Penetration. A failure of software or hardware controls could result in an exposure of information within the system. Such failures can involve the coupling of information from one user with that of another user. Software failure can render files or programs unusable, defeat or circumvent the security measures, or change, unintendedly, the security status of users, files, or terminals. The Hughes study discussed hardware failure for all post-1975 systems.

> The probability that an unintentional release of data due to hardware failure will occur is less than the hardware mean time between failures (MTBF) because parity check circuits will detect some circuit failures and others will not result in data release.[31]

Risk of software failure can be greatly reduced by carefully correcting errors and certifying the programs before implementation. Accidental disclosures may also occur by improper actions of machine operations in routing without deliberate intent. Anderson, in his computer security control study stated:

> . . . that the actual risk of classified information being made available to unauthorized persons due to misroute is quite small, and then only if the unclassified lines are continuously passively monitored for the eventuality. Note that the statement above does not suggest that misroute will not occur, only that its effect in an open-secure system is greatly exaggerated.[32]

The software controls in Chapter II are designed to reduce the probability of accidental penetration. Deliberate penetration from outside the system is already an acceptable risk level. As previously stated, physical, communication, administration, and personnel controls already meet stated DoD criteria. Personnel controls apply to only those individuals who have access to compartmented data.

The preceding sections have discussed the IDHS environment and have evaluated the risk. The comparison of the manual versus the automated security checks will aid in determining an acceptable risk level. Manual provisions are currently acceptable as secure in storing and processing compartmented intelligence.

Manual/Automated Procedural Comparison

This comparison will provide a method to judge the relative value of automated techniques. This section will discuss those techniques that are common to both types of systems, and those that are analogous.

In both systems, procedural techniques are used to provide physical protection for areas where compartmented data is used; to insure personnel possess appropriate clearance; to receive, control, disseminate, and access compartmented data; and to protect the compartmented data during transmission. The addition of automated techniques to increase the reliability of these procedures could be viewed as an attempt to increase the security of automated systems over that of manual systems.

> Analogous techniques used in the two systems are (1) data storage procedures, (2) data access procedures, (3) data access accounting, (4) storage check procedures, and (5) inventory procedures. Data stored in a manual system is protected by three combination safe locks with 125,000 possible combination codes. In an automated system, data storage is based upon access codes with 4,000,000,000 possible combinations in a normal 32 bit code word. The probability of data access through other than code word knowledge is considerably less in an automated system than in the present manual system.[33]

Access to data in a manual system is based on access lists and personal identification. In an automated system, access to data and files is based upon user/terminal profile tables and the requirement to submit the proper code word.

Accountability for data access in the manual system is performed by document sign-out. In the automated system, the access logs and security program reports can be reviewed daily, or more often if desired. Periodic inventory is used in the manual systems to insure documents have not been lost or stolen. The same can be accomplished in the automated systems. The files are periodically reviewed, the security logs examined, and the check sum totals used to insure data integrity. Table 2 summarizes the above techniques. The common techniques for storing and processing compartmented intelligence in a manual and automated system are:

a. Physical

b. Personnel

c. Communication

d. Administrative

It is evident that even with the use of modest security techniques in an automated system, a greater level of security is possible than with those techniques available in the manual system.

TABLE 2[34]

MANUAL/AUTOMATED SECURITY COMPARISON

MANUAL		AUTOMATED
Combination Locks	versus	Passwords and Access Codes
Access Lists	versus	User/terminal Profile Tables
Document Sign-Out	versus	File/Data Element Access Log
Daily Safe Check	versus	Access Log Review and Security Monitor Program
Periodic Inventory	versus	Record Counts, Check Totals, File Reviews

Summary

The IDHS FORSIG environment has been designed to prevent deliberate penetration from outside the system. Inadvertent or accidental disclosure of compartmented data has been identified as the problem.

The software controls discussed in Chapter II were designed to prevent accidental disclosure. The security kernel is the area where all security-related software, profiles, and lists are stored. Software must be verified correct and a system certified before an acceptable risk exists. Chapter IV, Conclusions and Recommendations, discusses verification and certification of the security software. Conclusions concerning ASSIST computer security at FORSIG are discussed and recommendations are made to bring the risk of penetration to an acceptable level.

CHAPTER IV

CONCLUSIONS AND RECOMMENDATIONS

Introduction

Since current personnel, administrative, physical, and communication security measures in force are approved by the Defense Intelligence Agency in processing compartmented data, the emphasis is placed on software security control techniques. The recommended software security controls do not depend on one control, such as a password, to provide security for the system. The recommended software incorporates a series of controls, each supporting the other as a security constraint, with all being subject to a security verification program. By having a series of controls, any one of which could deny access to a would-be-penetrator, the probability of penetration is reduced significantly. Although the probability is further reduced as more controls are added, at some point the additional controls are not cost effective.

Software security checks are designed to give a consistent verification of both identification and authorization levels of the individual user and terminal each time an attempt is made to access the system. The design of the software is such that it has the capability of detecting any accidental or intentional security breaches, identification of the time and person responsible for the breach, and the disconnection of that terminal.

The broad objective of finding ways to reduce the size and complexity of security-relevant software is a prerequisite to performing a convincing, logical verification that a system correctly implements the claimed access constraints when used with a good verification technique. Without such verification of correctness, a system cannot be considered an acceptable risk.

Verification

Both the design and implementation of software security controls in the security kernel must be tested to insure correctness. E. W. Dijkstra designed a process called structured programming. Dijkstra's design can be used in the verification of the operating systems' software.[35] The structure is a top down approach in which the program is built one level at a time. "At each level, the next lower level of the structure is denoted by a name (or abstraction) assigned to it. For each level, a proof, in which the denotation of each name denoting a lower level is considered to be correct, is constructed."[36] Each component of the operating system is constructed to be self-contained, and to operate correctly. The components are then able to communicate freely and without possibility of interference. Since each step of the construction process is proven to operate correctly, this constitutes a "proof" that the system as a whole will operate correctly. The programs that result from this process have a well defined structure and are practically error free because of proving correctness is proved at each level.

The verification process must be accomplished by individuals who are authorized access to compartmented intelligence. This is necessary because it is the software that will control access to compartmented data. It is virtually impossible to prove that the software is 100 percent correct, counteracting all threats under all conditions. It is also impossible to prove that all persons having access to compartmented data will not compromise this information in some manner. But DoD has accepted the condition that after an individual undergoes an expanded background investigation, the risk is acceptable. It is possible to test software for known threats and prove its correctness. Software properly structured, proven, and tested can also be certified as an acceptable risk.

Certification

Software certification involves proving that the security kernel is always invoked, is tamper resistant, and does validate each and every reference in the system.[37]

The system must be tested by expert technical personnel having access to the design and specifications of the entire system. Access to such material would not be in the hands of a potential penetrator because this information would be controlled as compartmented data. The strict controls are imposed because, the information on software and system specification is the heart of the protection features for the compartmented data contained in the system. "Certification should be performed by a group other than that responsible for the design, construction, or maintenance of an operational system."[38] This outside group of experts, given all the system's specifications, will design a plan of attack against the security constraints enforced by the software. A sufficient amount of time, approximately three months, should be given to plan the attack. After the attack is planned, the team of experts should initiate the attack on the site. After the attack is implemented with no successful penetration, the system should be certified as posing an acceptable risk level.

The concept of attempted system-penetration as a means of certifying a system is not new. "In 1970, a group of Rand researchers,

J. Anderson, R. Bisbey, and D. Hollingworth, demonstrated the practicality of system-penetration as a tool for evaluating the effectiveness and adequacy of implemented data security safe-guard."[39]

The thoroughness of the test is limited by the availability of manpower and money. The security kernel contains all software security functions necessary to minimize the certification process by localizing what has to be certified.

In order to accelerate the testing cycle and reduce the amount of manpower, the use of automated verification techniques which assist in the certification of the operating systems and application programs are summarized below.

1. Automatic analysis of the anatomy of an operating system, i.e., identifying all "testable segments" (sequence of code that has only one input and one exit) and all transfers between segments.
2. Quantifying the thoroughness of the testing by instrumenting the operating system to measure the fraction of segments and transfers exercised in each test and cumulatively over a series of tests.
3. Identifying the portions (segments and transfers) not tested in a series of test cases and indicating the input data needed to exercise them.
4. Identifying all entrances to sensitive areas of an operating system.
5. Identifying all interrupts and the logical paths they can initiate.
6. Investigating other characteristics of operating systems for suitability for automatic analysis and quantitative measurement, e.g., time depe dent processes.[40]

The test will determine the degree to which the system conforms to the security requirements. If any changes or modifications to the system occur, then the entire certification process must be conducted again. This is the only way an acceptable security mode of operation can exist.

Conclusions

The ASSIST problem, preventing disclosure of compartmented data to analysts who do not possess the necessary access, has been discussed by an examination of the Intelligence Data Handling Site (IDHS) at Forces Command Intelligence Group (FORSIG). Security in a multiuser, multilevel system is not an impossible task. However, the term "security" has to be modified to connote an acceptable level of security. No system is 100 percent secure, but a system may possess security features that reduce the risk of penetration to an acceptable level.

IDHS FORSIG has established and approved physical, personnel, hardware, administrative, and communications security measures and presently processes compartmented data in a system-high mode. To process in an open mode, software security must be implemented that would prevent users without a need to know from access to compartmented data. An open mode of operation allows all analysts to process simultaneously regardless of their level of access.

With one major exception, the security environment of the manual system used for processing compartmented and collateral data simultaneously is the same security environment as in an automated system. In the manual system the special security officer enforces security, while in the automated system the software enforces security constraints. The use of a special security officer is currently accepted as a means of providing an acceptable security risk. Software has not been accepted as an acceptable risk. The special security officer is human and thereby capable of mistakes. Software, properly constructed, reacts to each situation defined in the same manner each and every time, unless there is a failure within the software. But the software design is constructed to have checks and counter checks to prevent error. The probability of all controls failing at once, or going undetected, is extremely low. The software does not eliminate the need for a special security officer. The software is the special security officer's representative within the computer and performs the same functions as that of the special security officer in a manual system. The special security officer is still responsible for the security of compartmented data. The software is the tool that assists him in his duties.

The special security officer is responsible for receiving, storing, controlling, and disseminating compartmented data. The section on manual and automated comparison in Chapter II compares controls and shows that automated controls increase security. Security constraints governing compartmented data are contained in several DoD directives and can be placed in the software. Security is provided by installing sufficient software barriers/controls to prevent and/or detect penetration, or by requiring such a very large work factor to penetrate, that a would-be-penetrator would be detected by co-workers in the terminal area. The controlled physical environment surrounding the terminals and computer at FORSIG greatly aid in the detection of a deliberate penetration from within, although not eliminating the possibility of an attempt.

The various software control the areas of access, input/output, residual and audit trail, and also build upon the established physical, personnel, administrative, and communications security measures already at FORSIG to bring the risk of penetration to an acceptable level. But software is only as good as the individuals who design and construct it. It is therefore imperative that the software be carefully constructed and tested before implementation. After the constructing, testing, and implementing of software, the entire system must be tested by a team of technical experts to insure the system conforms to the security constraints placed upon it. The ASSIST software security controls design is listed and discussed in Chapter II and identifies the controls necessary for acceptable software security at FORSIG.

Once controls are designed, they must be protected from all users. The security kernel discussed in Chapter II offers an ideal location for software security controls. The kernel protects the software from being modified. The kernel, the center of the software security controls, ensures compartmented data is not compromised.

Research indicates that it is possible to build an adequately secured system for a particular operational environment. With properly structured software controls that have been subjected to an intensive and unsuccessful penetration attack, the IDHS FORSIG offers an existing environment that can be certified as possessing adequate security.

Recommendations

The following statements summarize the software recommendations regarding the multilevel security problem described in Chapter III at the ASSIST intelligence data handling site:

1. Implement software security controls listed on pages 12 and 13. The software must be constructed carefully in order to prevent errors in the programs that could defeat the constraints to be enforced by the software.

Access controls are the first software controls the user encounters. They are of primary importance because these controls identify who, where, and what access is authorized throughout the system. Special emphasis should be given to controls numbered one, two, and ten, on pages 12 and 13. These controls identify the user/terminal and establish the access rights. Correct identification of the user/terminals is of prime importance because it is the location of the terminal, and the user's clearance that governs the access rights.

After the access rights are identified, they accompany each transaction and restrict access to only that data authorized. Input/output controls insure that the user performs only those operations authorized. Once the process is complete, I/O controls release data to only those user/terminals processing the necessary clearance level. The main emphasis should be placed on input/output control numbers two, three, five, seven, and eight on page 13. These numbered designs are concerned with classification and release of classified (compartmented) data to those users not appropriately cleared.

The control of residual data is necessary because many users are assigned the same memory space. If this space is not erased there is a possibility a user may gain access to data to which he is not authorized.

The audit trail is used to verify that the system is operating correctly and that it is being used properly. The system must be able to identify all attempted violations, accidental or deliberate. All the logs in Table 2, page 17, are necessary to provide the system security officer and special security officer with sufficient information to insure continuing security.

Special emphasis should be given to design number five under audit trail, page 13. The internal software and hardware checks would be a means of verifying their continuing correct operation. The check would be inserted into the system as a program imitating a user. The test program should attempt to violate security controls, and then verify that the system gives the correct response in each case. The program must communicate with the system as a normal user would. Communications of this type would require that the program be routed to a remote location, and back to the computer again. The program routing would also check channel controls. Every time the test program violates security constraints, the system security officer must be notified and the system's operation terminated until the problem can be resolved.

Implementation of software controls at IDHS FORSIG will enhance the security posture of that ASSIST site. Extreme care must be taken in constructing and testing the designed software.

2. Incorporate software security controls into a security kernel. The objective of the security kernel is to integrate all security related functions into one part of the operating system. The collection of all security functions into a central area aids in the protection and verification of their correctness. The security kernel has initial control over queries and transactions, and every user is forced to rely upon it. The security kernel enforces the security constraints on the use of files and the release of data. The functions of the kernel are listed on page 20. It is the performance of these functions that insures that the security constraints are enforced. To insure that the kernel is able to perform properly, two conditions must exist. First, the security kernel must be tamper-proof, allowing only authorized personnel to modify or alter its functions. Second, the kernel must always be invoked. No method should be open for a user to bypass the kernel. Presently, the kernel is hardware dependent and the appropriate hardware is in service at the IDHS FORSIG.

3. Attempt to penetrate the system as a means of certifying that

the software possesses an acceptable security risk. The FORSIG system is presently certified to process compartmented data in a system-high mode. After software security controls are implemented at FORSIG, a method of proving that the system enforces security constraints must be accomplished before certification. By providing a team of experts with all the possible data on the system, and allowing them to attack the system with their program, an advantage is created which would normally not exist for a would-be penetrator. It must be emphasized that a change or modification of the software would void the certification.

This thesis has identified what is considered to be the minimum number of software security controls, and verification/certification techniques necessary to prove their correctness, for the intelligence data handling site at Fort Bragg, North Carolina. The total ASSIST environment--physical protection of the facilities, personnel access controls to the terminals, limited locations of nonappropriately cleared terminals, encryption of communication lines, along with the software security techniques recommended in this thesis--contributes to reduce the risk of penetration and bring this risk to an acceptable level.

ENDNOTES

1. Willis H. Ware, (ed.), Security Controls for Computer Systems, (Santa Monica, California: The Rand Corporation), p. xi.

2. Department of Defense, ADP Security Manual, (Washington, D. C.: Secretary of Defense), p. 1.

3. Ibid.

4. Ware, op. cit., p. 35.

5. Ibid., p. 39.

6. Defense Intelligence Agency, Security of Compartmented Computer Operations (U) Confidential, (Washington, D. C.: Headquarters Defense Intelligence Agency), p. 7.

7. TRW Systems Group, Army System for Standard Intelligence Support Terminals (ASSIST), (1976), p. 3.

8. Ibid., p. 6.

9. Department of Defense, Special Security Manual (U) Top Secret, (Washington, D. C.: Secretary of Defense); see also, Department of Defense, ADP Security Manual, (Washington, D. C.: Secretary of Defense); see also Defense Intelligence Agency, Physical Security Standards for Sensitive Compartmented Information Facilities, (Washington, D. C.: Headquarters DIA); see also, Department of Defense, Communication Security (Washington, D. C.: Secretary of Defense).

10. IBM Corporation, The Considerations of Data Security in a Computer Environment, (White Plains, New York: DP Division), p. 10.

11. Ibid., p. 16.

12. Department of Defense, ADP Security Manual, op. cit., p. 9.

13. Department of Defense, Special Security Manual, (Washington, D. C.: Secretary of Defense); see also, Department of Defense, Physical Security Standards for Sensitive Compartmented Information Facilities, (Washington, D. C.: Secretary of Defense.

14. Ware, op. cit., p. 48.

15. James P. Anderson, AF/ACS Computer Security Controls Study, (Fort Washington, Penn.: Anderson and Company), pp. 8-33; see also, Anderson, Computer Security Technology Planning Study, Vol. 2, (Fort Washington, Penn: Anderson and Company), pp. 6-54; see also, Ware,

op. cit., pp. 26-62; see also, J. T. Shen, Multilevel Security for Computer System Networks: A Survey and Discussion, (San Diego, Ca.: Naval Electronics Laboratory Center), pp. 4-22; see also, Hughes Aircraft Company, Security of the TACC Data Base Study (Description of Automatic Data Base Security Techniques), (Fullerton, Ca.: Hughes Aircraft Company), pp. 2-1, 5-28; see also, Gerald P. Popek, Access Control Models, (Cambridge, Mass.: Harvard University), pp. 114-137; see also, Scitek, Inc., Report on the Design Research of an Advance Operational Information System, (State College, Penn.: Scitek, Inc.), pp. 20-36; see also, M. Gasser, Design of a Secure Communication Processor-Input/Output Processor, (Bedford, Mass.: Mitre Corporation), pp. 1-14; see also, R. Bisbey, D. Hollingworth, Protecting Errors in Operating Systems: Allocation/Deallocation Residuals, (Marina delRay, Ca.: Information Science Institute), pp. 1-13; see aiso, IBM, op. cit., pp. 1-33.

16. Ibid.

17. Turn, Rein, and Ware, "Privacy and Security in Computer Systems" American Scientist, (March-April, 1975), p. 2.

18. W. L. Schiller, Design of a Security Kernel for the PDP 11/45. (Cambridge, Mass.: Mitre Corporation), p. 64.

19. Ibid.

20. Department of Defense, Computer System Security Course Manual, (Washington, D. C.: Computer Institute), p. 53.

21. Michail D. Schroeder, Honeywell Information Systems, Inc., Security Kernel Evaluation for Multics, (Cambridge, Mass.), p. 4.

22. Ibid., pp. 4-5.

23. James P. Anderson, Computer Security Technology Planning Study, (Fort Washington, Penn.: James P. Anderson and Company), pp. 6-54; see also, Schroeder, op. cit., pp. 1-15; see also, Schiller, op. cit., pp. 1-69.

24. Ibid. 25. Ibid. 26. Ibid.

27. Hughes Aircraft Company, Security of the TACC Data Base Study (Description of Automatic Data Base Security Techniques), (Fullerton, Ca.: Hughes Aircraft Company), pp. 2-1, 5-28.

28. Edward K. Yasaki, "A New Science Emerges: Plugging the Holes in the Operating System," Datamation, (February, 1974), p. 92.

29. TRW Systems Group, Army System for Standard Intelligence Support Terminals (ASSIST), 1976.

30. James P. Anderson, AF/ACS Computer Security Controls Study, (Fort Washington, Penn.: Anderson and Company), p. 29.

31. Hughes Aircraft Company, op. cit., p. c-1.

32. Anderson, op. cit., p. 4.

33. Hughes Aircraft Company, op. cit., pp. 6-13.

34. Ibid., pp. 6-17.

35. Anderson, op. cit., p. 49.

36. Ibid. 37. Ibid., p. 48.

38. Willis H. Ware, (ed.), Security Controls for Computer Systems, (Santa Monica, Ca.: The Rand Corporation), p. 39.

39. R. Turn, and others, A Brief History of Computer Privacy/Security Research at Rand, (Santa Monica, Ca.: The Rand Corporation), pp. 4-5.

40. Anderson, op. cit., pp. 49-50.

BIBLIOGRAPHY

Books

IBM. The Considerations of Data Security in a Computer Environment. White Plains, New York: IBM Corporation, DP Division, 1976.

Tassel, Dennis Von. Computer Security Management. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1972.

Turn, R. A Brief History of Computer Privacy/Security Research at Rand. Santa Monica, California: The Rand Corporation, March 1972.

Government Documents

Anderson, James P. AF/ACS Computer Security Controls Study. 19034, Fort Washington, Penn.: James P. Anderson and Company, for Command and Management System Headquarters, Electronic Systems Division, AFSC, November, 1971.

________. Computer Security Technology Planning Study. Vol. 2, Fort Washington, Penn.: James P. Anderson and Company, for Electronic Systems Division, AFSC, 1972.

Ballew, Glynn E., Boettcher, Charles W., Jr., Hatfield, Kenneth H., Padgett, Robert L. DIALOS Enhancement. McLean, Virginia: Tesdata Systems Corporation, Analytical Services Division, for Rome Air Development Center, Griffis Air Force Base, New York, 1973.

Bell, D. Elliott. Secure Computer Systems: Mathematical Foundations. Vol. I, Bedford, Mass.: Mitre Corporation, for Electronic Systems Division, November, 1973.

________. Secure Computer Systems: A Mathematical Model. Vol. II, Bedford, Mass.: Mitre Corporation, for Electronic Systems Division, November, 1973.

________. Secure Computer Systems: A Refinement of the Mathematical Model. Bedford, Mass.: Mitre Corporation, for Electronic Systems Division, AFSC, 1974.

Carlstedt, Jim. Protection Errors in Operating Systems: Validation of Critical Conditions. Marina delRay, California: Information Science Institute, for Defense Advance Research Projects Agency, Arlington, Virginia, 1976.

Deerfield, A., Durgin, F., Tanenbaum, R. Final Report for SPE/AADC Interface Study. Bedford, Mass.: Raytheon Company Missile System Division, for Naval Research Laboratories, Washington, D. C., February, 1974.

Defense, Department of. ADP Security Manual. DoD 5200.28M, Washington, D. C.: Assistant Secretary of Defense, January, 1973.

———. Computer System Security Course Manual. Washington, D. C.: Computer Institute, 1974.

Defense Intelligence Agency. Physical Security Standards for Sensitive Compartmented Information Facilities (FOUO). DIAM 50-3, Washington, D. C.: Headquarters Defense Intelligence Agency, 31 July, 1976.

Gasser, M. Design of a Secure Communications Processor - Input/Output Processor. Project 7210, Bedford, Mass.: Mitre Corporation, for Directorate of Planning and Technology, Electronic Systems Division, AFSC, 1972.

Gerhard, William D. Proceedings of Invitational Workshop on Network of Computers. NOC-68, Fort George G. Meade, Maryland: National Security Agency, September, 1969.

Hollingworth, Dennis, Bisbey, Richard, II. Protection Errors in Operating Systems: Allocation/Deallocation Residuals. Marina delRay, California: Information Science Institute, for Defense Advanced Research Projects Agency, Arlington, Virginia, 1976.

Hughes Aircraft Company. Security of the TACC Data Base Study (Description of Automatic Data Base Security Techniques). Fullerton, California: Hughes Aircraft Company, Ground Systems Group, for Electronic Systems Division, AFSC, September, 1970.

Popek, Gerald J. Access Control Models. Cambridge, Mass.: Harvard University, Center for Research in Computing Technology, for Command and Management Systems Headquarters, Electronic Systems Division, AFSC, 1973.

Schacht, J. M. Jobstream Separator: Supportive Information. Bedford, Mass.: Mitre Corporation, for Electronic Systems Division, AFSC, January, 1976.

Schiller, W. L. Design of a Security Kernel for the PDP 11/45. Bedford, Mass.: Mitre Corporation, for Electronic Systems Division, AFSC, 1973.

Schroeder, Michael D., and Honeywell Information System, Inc. Security Kernel Evaluation for Multics (Interim Report). MIT-Project MAC, Cambridge, Mass.: for Federal Systems Operation, McLean, Va., 1975.

Scitek, Inc. Report on the Design Research of an Advance Operational Information System. Technical Report 102-2, 29, State College, Pa.: Scitek, Inc., for Office of Naval Research, February, 1972.

Shen, J. T. Multilevel Security for Computer System Networks: A Survey and Discussion. San Diego, California: Naval Electronics Laboratory Center, May 7, 1974.

Stryker, David. Subversion of a 'Secure' Operating System. NRL Report 2821, Washington, D. C.: for Naval Research Laboratory, May, 1974.

Ware, Willia H. Security Controls for Computer Systems. Santa Monica, California: The Rand Corporation, 1970.

Journals and Magazines

Babcock, J. D. "A Brief Description of Privacy Measures in the RUSH Time Sharing System," Proceedings of the 1967 Spring Joint Computer Conference, AFIPS, Vol. 30.

Beardsly, Charles W. "Is Your Computer Insecure?" IEEE Spectrum, 9:1, January, 1972.

Butler, Robert W. "Computer Fraud: Latest in White-Collar Crime," The Kansas City Times, 109:8, (Kansas City, Missouri), September 16, 1976.

Chacon, Jose A. "Computer Security," Perspectives in Defense Management, Autumn, 1973.

"Classification Management," Journal of the National Classification Management Society, IV:2, 1968.

"Classification Management," Journal of the National Classification Management Society, Vol. IX, 1973.

"Classification Management," Journal of the National Classification Management Society, Vol. XI, 1975.

"Computer Crime Tops $2 Million," Army Times, May 24, 1976.

"Computer Security ...The Imperative Nuisance," Infosystems, February, 1974.

Conway, R. W., Maxwell, W. L., Morgan, H. L. "Selective Security Capabilities in ASAP - a File Management," Proceedings of the 1972 Spring Joint Computer Conference, AFIP, Vol. 40.

David, Heather M. "Computers, Privacy, and Security," Computer Decisions, May, 1974.

Denning, Peter J. "Third Generation Computer Systems," Computing Surveys, Vol. 3, No. 4, December, 1971.

Feistel, H. "Cryptograph and Computer Privacy Protecting Personal Data Banks," Scientific Digest, Vol. 228, May 1973.

Graham, G. S., Denning, P. J. "Protection - Principles and Practice," Proceedings of the 1972 Spring Joint Conference, AFIP, Vol. 41, 1972.

Hoffman, L. J. "Computers and Privacy: A Survey," Computing Surveys, Vol. 1, No. 2, June, 1969.

Kingsley, Norman. "Computers Can Be Subverted," Marine Corps Gasette, 58:53, February, 1974.

Lampson, B. W. "Dynamic Protection Structures," Proceedings of the 1969 Fall Joint Computer Conference, AFIP, Vol. 35.

Parker, Don B. "Computer Security: Some Easy Things To Do," Computer Decisions, January, 1974.

Porter, W. Thomas, Jr. "Computer Raped by Telephone," The New York Times Magazine, September 8, 1974.

Rapoport, R. "Electronic Alligators," Saturday Review of Sciences, 1:35-8t, March, 1973.

Schimke, W. 'Modern Operations - Linked Data Processing and its Military Applications," Militaer Technik, No. 2, 1972.

Slaughter, John B. "Understanding the Software Problem," Proceedings of the 1974 National Computer Conference.

Tassel, Dennis Von. "Information Security in a Computer Environment," Computers and Automation, July, 1969.

Turn, Rein, and Ware, W. H. "Privacy and Security in Computer Systems," American Scientist, March-April, 1975.

Weiss, Harold. "Computer Security an Overview," Datamation, January, 1974.

Weissman, C. "Security Controls in the ADEPT-50 Time Sharing Systems," Proceedings of the 1969 Fall Joint Computer Conference, AFIPS, Vol. 30.

Yasaki, Edward K. "A New Science Emerges: Plugging the Holes in the Operating System," Datamation, Vol. 20, No. 2, February, 1974.

Pamphlet

TRW Systems Group. Army System for Standard Intelligence Support Terminals (ASSIST), Redondo Beach, California, 1976.