HUDCAPS Access Controls Need Improvement

Audit Report

September 30, 1997

U.S. Dept. of Housing & Urban Development
Office of Inspector General
Office of Audit
Information Systems Audit Division
451 Seventh Street, S.W.
Washington, DC 20410

97-DP-166-0001

U.S. Department of Housing and Urban Development
Washington, DC 20410-4500

OFFICE OF INSPECTOR GENERAL

MEMORANDUM FOR: Steven M. Yohai, Chief Information Officer, and
Director, Office of Information Technology, AMI

FROM: Benjamin K. Hsiao, Director, IS Audit Division, GAA

SUBJECT: Audit Report, HUDCAPS Access Controls Need Improvement

Attached is a draft report on our review of the access controls of the HUDCAPS system. In accordance with our regular practice, we are forwarding it to you for your review. Please provide any comments you may have to me by August 15, 1997.

We conducted the review to evaluate the access controls over the HUDCAPS financial application. Our objectives for this review were to determine if the three mainframe support systems, CA-Top Secret, DB2 and CICS, were installed and configured with sufficient controls to ensure that the HUDCAPS application was adequately protected against unauthorized access.

Officials who receive draft OIG reports for review and comment are required to use due care to avoid premature or otherwise improper disclosure of the information they contain. Drafts may not be released outside of the Department without approval of the Assistant Inspector General for Audit.

Thank you for the assistance provided to us by your staff during the course of our review. Should you have any questions or require additional information, please contact me at 708–3444, ext. 149.

Attachment

Table of Contents

Access Controls Over HUDCAPS Need Improvement

The purpose of this audit was to evaluate the general access controls over the HUD Central Accounting and Program System (HUDCAPS) on the Hitachi mainframe computer. Specifically, our objectives were to determine whether the three software support components, **CA-Top Secret, DB2** and **CICS,** were installed and configured with sufficient controls to ensure that the HUDCAPS application was adequately protected against unauthorized access. CA-Top Secret is Computer Associates' security software used to control access to the Hitachi mainframe; DB2 is the IBM data base management system where HUDCAPS' data is stored; and CICS is the IBM system software used to support on-line users of HUDCAPS.

As a result of our review, we found that the Department has not taken adequate measures in the general control environment during HUDCAPS' implementation to control system access. This situation has exposed the Department to the risk of unauthorized use of restricted data and programs. We base the above finding on the implementation deficiencies within each of the three software components. The summary description of each software support component along with its deficiencies is given below.

CA-Top Secret Security software controls access to Hitachi mainframe resources (software, data, terminals, storage devices, etc.) by limiting how, when and which resources a user can access in the system. If the settings are improperly configured, unauthorized users can gain full-access (read, update, create, delete) to any software and data. We found that there was no resource access-authorization checking for some users; and that not all facilities were in FAIL mode, which is the highest security mode. Both conditions expose HUDCAPS data and software to improper use and/or modification.

DB2 Data Base Management software organizes HUDCAPS data, as well as, provides its own access security over HUDCAPS. If the components of DB2 are not properly configured, any DB2-organized data on the mainframe can be placed at risk of unauthorized use. We found that the highest authorization level was granted to one user, which is not the recommended procedure. Further, this user was a contractor employee whose access activities were not monitored. We also found that the granting of DB2 access privileges for job-related user groups was not controlled. Both deficiencies result in a risk that access can be granted to individuals who then can make unauthorized changes to HUDCAPS data and software. The main reason for this control deficiency is a lack of a formal process for granting DB2 access.

CICS Customer Information Control System software is designed to support the development and processing of on-line, interactive applications by providing pre-defined screen formats, telecommunications routing and other control information. CICS concurrently handles the transmission to and the update of HUDCAPS data base by user-entered transactions from different workstations. We found that the access to three powerful vendor provided CICS transactions were granted to a large number of users (over 70). Granting an excessive number of users access to these transactions increases the risk of unauthorized processing of HUDCAPS data, and/or system failures. These transactions are:

- CEMT - a master transaction used to changes CICS operating parameters;
- CEDF - a transaction used to test on-line software;
- CEDA - a transaction used to change CICS system tables and interactively define and add new resources.

Recommendations

We recommend that the Director, Office of Information Technology:

- Reconfigure the CA-Top Secret security software to operate at global Fail Mode. Also, ensure that all system facilities are in Fail Mode;
- Control the use of the bypass privilege for accessing HUDCAPS software and data;
- Adopt a formal process to control the granting of DB2 access privileges;
- Restrict the use of the CEMT transaction to a minimal number of system operators and the system programmers; and remove CEDF and CEDA transaction access from all non-CICS system programmers.

Department Comments

We provided the draft report to the Office of Information Technology (IT) on July 16, 1997. We received written comments on September 2, 1997. IT generally did not agree with the recommendations of our audit. Their comments and our response are provided in Appendix C.

Introduction

Background

We reviewed the access controls of the HUD Central Accounting and Program System (HUDCAPS), one of HUD's major financial systems. HUDCAPS incorporates a commercial off-the-shelf (COTS) software product called the Federal Financial System (FFS), marketed and maintained by American Management Systems (AMS). Our purpose in reviewing access controls was to provide reasonable assurance that information processed is properly safeguarded. Access controls have a significant impact on the overall security of application systems. Access controls can be placed either within the application itself, or in the general control environment. This general control environment exists outside of the application itself. If general controls are weak, they may invalidate controls built into the application itself and expose the application system to serious risks, such as jeopardizing the integrity of data and unauthorized disclosure of sensitive information.

Since HUDCAPS runs on the Department’s Hitachi mainframe computer system at Lanham, Maryland, the procedures and systems supporting that mainframe environment can have a major impact on HUDCAPS’ operations. Therefore, attention must be paid to ensure that these systems and procedures are properly controlled. These procedures and systems include the operating system of the mainframe computer, the procedures for controlling changes to application programs, software development processes, communication, data base management, on-line transaction management, the physical security over the operations center, and access security over the applications running on the mainframe. For this latter purpose, the access security over the applications running on the mainframe, there is specialized security software installed on the system.

For our review, we concentrated our efforts in evaluating access controls contained within the general operating environment, and not within the HUDCAPS application itself. More specifically, we evaluated three key system software applications that support the operations of the HUDCAPS program. We evaluated the data base management system, DB2, used by HUDCAPS. We also reviewed the on-line transaction control system called the Customer Information and Control System (CICS). Finally, we looked at the mainframe security software, CA-Top Secret.

DB2 is IBM's relational data base management system. It is a computer-based record keeping system whose purpose is to record and maintain information. DB2 data is maintained separately from the application and stored in a common repository that can be shared by different programs and applications.

CICS is software that facilitates the development of on-line, interactive applications, where screen formats, terminal routing and other control information are provided by CICS. Under the CICS facility, HUDCAPS operates as a transaction-processing system.

CA-Top Secret is a commercially available security package, which, if properly installed, can protect all resources of the mainframe computer from unauthorized access.

Audit Objectives

Our objective for this review was to determine if the three mainframe support systems, DB2, CICS, and CA-Top Secret, were installed and configured with sufficient controls to ensure that the HUDCAPS application was adequately protected against unauthorized access.

Audit Scope and Methodology

We reviewed DB2 implementation and maintenance to ensure that the integrity of the DB2 system had not been compromised by the addition of unauthorized programs, and an adequately secure DB2 operating environment has been established for running the production HUDCAPS application. We also reviewed existing DB2 documentation to determine if it adequately described the DB2 system, features, enhancements and HUDCAPS data base administration.

We also evaluated CICS implementation and maintenance to determine the adequacy of security over HUDCAPS production on-line environments. Specifically, we reviewed the setup of the CICS region to determine whether the region supporting HUDCAPS was adequately protected. We also reviewed the controls over powerful CICS transactions to determine whether these transactions are protected from unauthorized access. Additionally, we reviewed the adequacy of security over CICS system libraries, CICS audit trails and approval processes for CICS system changes. We further determined whether CICS system transactions have adequate security.

Finally, we reviewed the installation and configuration of the CA-Top Secret security software. Our objective was to ensure that the HUDCAPS software and data were adequately secured. We did this by analyzing how critical and sensitive files were protected within the HUDCAPS application.

Audit Period

We performed our field work from September 1995 through March 1997. We reviewed procedures and documentation from the period October 1994 through December 1996. We conducted our audit in accordance with generally accepted governmental auditing standards.

Access Controls Over HUDCAPS Need Improvement

The Department has not taken adequate measures during HUDCAPS' implementation to control access to the system. We found that the security software, CA-Top Secret, is not operating in the full protection mode; access to HUDCAPS data within the data base, DB2, is not adequately controlled; and an excessive number of users have been granted access to powerful CICS transactions. These control deficiencies have exposed the Department to the risk of unauthorized changes of restricted data and programs, and/or system failures.

Control over access to the HUDCAPS system is determined by a combination of factors, both internal and external to the program. Internal controls are built into the HUDCAPS application itself. External controls are part of the environment in which it operates. These external controls are found in the mainframe computer's hardware, its system software, and in procedures for managing operations. For our review, we focused on the external control environment. Specifically, we looked at the controls within three of the mainframe computer's system software programs (see the following

Access Control Environment

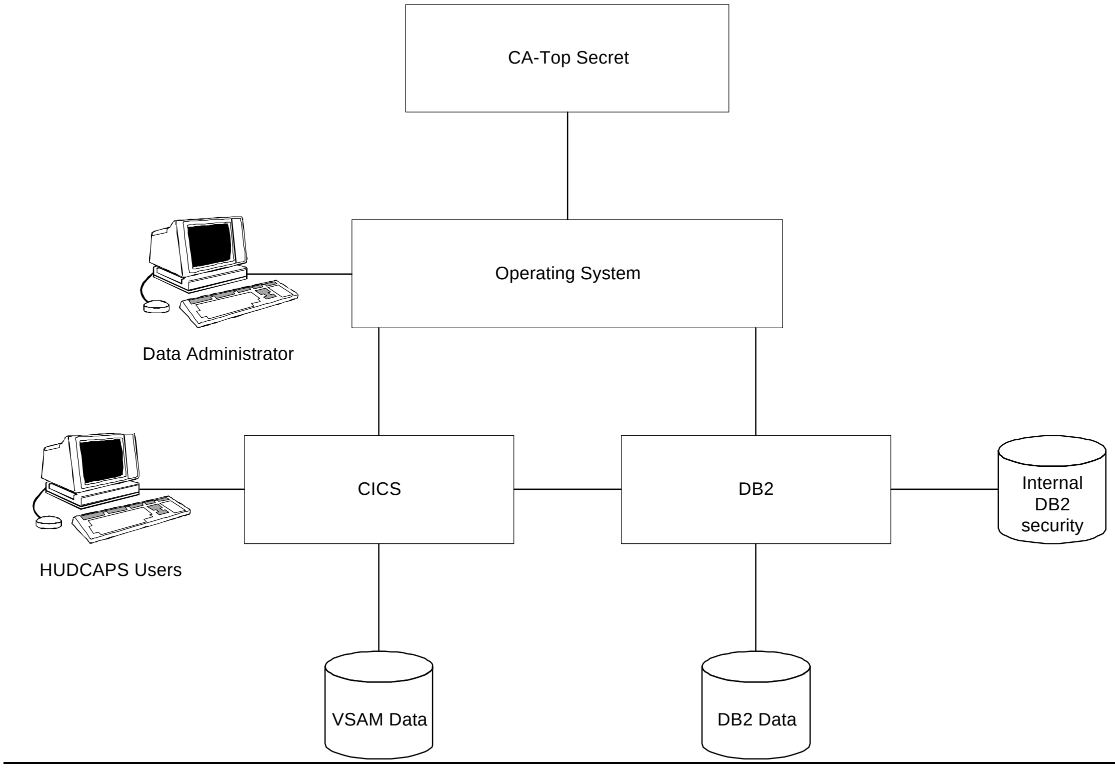


diagram).

One of the software programs that we reviewed is called CA-Top Secret. This program is a general mainframe security package that provides access control to protect data from accidental or deliberate destruction, modification, disclosure and/or misuse. It allows for the control of DB2 data base resources, and enhances and simplifies internal DB2 security administration. It is capable of logging, reporting and on-line monitoring of all access activities.

A second software program that we examined is the IBM relational data base management system (DB2). This is the software that controls the organization, storage, retrieval, security and integrity of data in a data base. It accepts requests from an application, such as HUDCAPS, and instructs the operating system to select the appropriate data for update.

The third software program that we reviewed is called the Customer Information Control System (CICS). It controls the interaction between the HUDCAPS application and users. It does this by giving the application programmers the capability of creating automated data entry and retrieval formats, which appear on users' personal computer screens. These formats simplify the interaction between the users and the HUDCAPS application. CICS also provides routing, password security, transaction logging for error recovery, and activity journals for performance analysis.

CA-Top Secret: Access Control Software

CA-Top Secret is an access control security package sold by Computer Associates International, Inc. (CA). It is an hierarchical approach starting from the highest level of the computer environment. When properly implemented, CA-Top Secret is designed to protect against unauthorized access and to permit users to perform only authorized functions. CA-Top Secret controls access to resources by limiting system entry and limiting how, when, and which resources a user can access in the system. CA-Top Secret is now installed on the Department's Hitachi mainframe computer.

CA-Top Secret controls access to a system by first identifying users through an **Ac**cessor **ID** (ACID). ACIDs (or userids) can be assigned to an individual or groups of individuals such as departments or divisions. When a group of users needs to use a set of identical resources in the same way, it is convenient to define this set of access authorizations once and then associate the entire set with each of the users in the group. Regions or areas of the mainframe's on-line facilities (known as the Customer Information Control System, CICS) processing HUDCAPS production data itself is assigned an userid. In that way, all programs running in a particular CICS region are further restricted access to other programs and data depending on the access permission granted to the userid assigned for that region. A more detailed discussion of CA-Top Secret and CICS are contained in Appendices 1 and 2, respectively.

❑ Bypassing Dataset Security Check

As described in Appendix 1, system resources are secured by first defining them to CA-Top Secret, and then by providing access authorizations for each userid that will need to use those resources. Authorization is the permission required to access a particular resource. In certain unusual circumstances special so-called *bypass* attributes can be added to the userid thereby granting it special privileges in accessing resources. With this bypass attribute added, any time a particular userid makes a certain type of access request, those requests will bypass security checks. The guidance given on the use of this attribute by CA-Top Secret states: "This type of 'blank check' for security access is not recommended for ordinary usage but may be considered when setting up disaster recovery procedures (where security needs to be bypassed in a hurry)." The guide goes on to state: "This type of action is not recommended for most users...bypass options (also called 'no-check attributes') should be used with great care if at all."

We found that the userid assigned to the HUDCAPS CICS region was granted the privilege of bypassing security data checks or NODSNCHK. With this privilege, programs running in this CICS region can gain access to all datasets throughout the mainframe system with no security check. There is no sound reason for this userid to have access to all the datasets on the mainframe computer.

We also determined that systems programmers and DB2 administrators have the ability to submit batch jobs under the authority of the userid assigned to the HUDCAPS CICS production region. Because this userid has been assigned the NODSNCHK, or bypassing dataset access checking, these individuals are able to update all datasets on the production system.

❑ Department Slow to Initiate FAIL Mode

To facilitate its installation, CA-Top Secret has four different modes of operation. Each mode gives increasing control over access to resources. The four modes are *Dormant*, *Warn*, *Implement*, and *Fail*. CA-Top Secret is initially installed in the Dormant mode. In this mode, CA-Top Secret does not validate user requests nor normally protect resources. The next level of operation, Warn mode, gives administrators an opportunity to adjust the system by examining security reports for undesired results before imposing security restrictions on users. Violations do not result in failing requests, so processing activity is not affected. Under Implement mode, CA-Top Secret is active and will fail any unauthorized request from a defined user. Users not defined to CA-Top Secret execute normally, but cannot access protected resources. In Fail mode, CA-Top Secret is in full control of access requests. All users must be defined and all defined resources are protected. Access requests not conforming to an existing access rule will fail. Fail mode is the vendor recommended option for a full production mainframe system such as the one supporting HUDCAPS.

Although CA-Top Secret was installed on November 6, 1993, at the time of our review, three years later, we still found that not all of the systems' facilities and resources were protected under the Fail mode, which is the highest security level. We identified three operating system facilities used to control job processing, program initiation, and memory management that were still in the Implement mode and one facility in Warn mode. Although the three facilities have since been placed in Fail mode, this situation represented a serious exposure. To be fully protected, it is necessary that HUD's Hitachi production mainframe be in full Fail mode for all facilities. Without Fail mode, production data, and system software products would not be fully protected under CA-Top Secret. Consequently, knowledgeable perpetrators could gain unauthorized access and cause system failures or loss of data.

In another instance, we found one of the areas within CICS was operating under CA-Top Secret in the Warn mode. We were able to establish that this situation lasted at least from October 1994 to June 1995. Although this deficiency has since been corrected, this represented a serious exposure. Up to June 1995, all HUDCAPS software developers were allowed to execute unauthorized programs through this CICS function which could update production files. This condition should never be allowed to occur again. ADP Security of IT must maintain segregation of duties between software developers and production (operations). This separation is essential to prevent unauthorized activities that could modify or destroy critical data and disrupt the continuity of system operation.

DB2 : Data base Management Software

DB2, the data base management system, plays an integral part in the administration of an application's data. Among other things, it organizes data and can, if setup properly, offer some level of security over access to a system's information and resources. If the components of DB2 are not properly configured, not only the user's data, but other's data as well on the same computer can be put at risk of unauthorized use.

The DB2 system allows users and system administrators to perform various functions or operations over the system and HUDCAPS data. These functions include defining data, manipulating data, and granting and revoking access to data. Individuals or groups of individuals are granted privileges to perform specific functions based on the level of access authority assigned to them by the data base administrator. These privileges will vary based on the need to perform certain tasks. Security is imposed on the system in the form of unique identifiers (ID's) to prevent individuals or groups from performing functions or accessing data to which they have no rights.

DB2 controls access to its functions and data through a system of hierarchical levels of authorities. People or groups are assigned to a level of access authority. Each level of access has its own rights and privileges in the system. The higher the level, the more rights are assigned. Some of these authorities at the highest levels give very broad sweeping rights and, therefore, must be particularly well controlled.

❑ No Control over the Highest Authorization Level

When a new application such as HUDCAPS is installed on the DB2 system, the data base administrator must insure that proper access authority is assigned to individuals and job-related groups.

Since the access authority is hierarchical in nature, the highest level must be assigned first. This first level can be assigned to one or two ID's. The ID's can be of an individual or of a generic position such as "Data Base Administrator." This person then has the authority to further designate access authority and system rights to other users of the application at lower levels than his/her level. These other users, once given the first level rights can, in turn, grant equivalent first level access authority down the hierarchy structure.

This highest level of authority, which is called the Installation System Administrator Authority (INSTALL SYSADM), is very powerful, given the broad level of rights and privileges assigned to it. This person has authority to access all DB2 resources and issue all DB2 commands. Unless this position is properly assigned and controlled, there is potential for great risk to the application, if the person assigned should abuse this authority.

The best practice in setting up the INSTALL SYSADM authority is to assign it to a generic ID and not to a specific individual. The ID then is secured and controlled. However, we found that INSTALL SYSADM authority was assigned to an individual. Further, this individual was a contractor employee whose activities were not normally recorded or tracked.

The risk inherent in this situation can be minimized if proper precautions are taken. No individual should be assigned this authority directly through his/her personal userid. This position should be assigned to a generic ID. The password for this ID should be kept in a secure, locked location, controlled by the system security officer. When it becomes necessary to use this ID again, usually only in an unusual situation after the initial system set-up, a report of the situation is prepared and the password in retrieved from the secured location. After the incident, a new password would be assigned and put back in the secured place.

❑ No Control Over Access Granting Authority

Users of applications such as HUDCAPS normally should have access to only their own programs and data. System utilities may be called upon to perform certain functions in support of the users programs, but under no circumstances should users be able to directly access the files containing the system utilities and programs. The potential for serious disruption to the DB2 system itself is great should a non-authorized user enter these areas.

We found during our review that numerous non-system programmers were granted the authority to read, write, and delete the DB2 system programs. These system programs were being used by the HUDCAPS application software. We discovered this weakness by identifying the names of the DB2 system programs being used. We then used CA-Top Secret reports to see to whom and what kind of access was granted.

We also found no access controls over user groups. DB2 has the ability to create groups to which access level privileges are assigned. Individuals assigned to the group can exercise the authorities granted to the group regardless of their own individual access level. However, individuals assigned to groups should individually have at least as high an access level as the group's. We found that HUDCAPS administrators did not have any documentation or formal method in place to support the assignment of individuals to groups.

Contributing to this situation, we found that there is no formal process to control granting access privileges. Whenever DB2 access grants are issued, there is no formalized request, no formal approvals, or documentation to log the access granted. This condition results in a risk that access can be granted which allows an individual to perform unauthorized changes to production resources.

❑ The DB2 Audit Trail Is Disabled

One of the features that DB2 offers is a log of all transactions and resources used by a given application. This feature is called the *Audit Trail*. The Audit Trail is an extremely effective tool for monitoring activity within the computer. It can track a variety of operations, depending on how it is set up. For example, it can be set to monitor access attempts denied by DB2; the results of granting and/or revoking access privileges; feedback from actions creating, altering, or deleting tables of data; and assignment and changes to access identification numbers. We found the Audit Trail feature turned off.

Without the Audit Trail, HUD's management would lose the ability to monitor changes made in granting user access privileges, and whether they are done in an authorized manner. Additionally, without a report of attempted and denied accesses, management would lose an indicator either for training needs or for unauthorized attempts to penetrate the system. The use of the Audit Trail function would be of particular help for HUD management in the case of the HUDCAPS system, because this system is administered by contract personnel. The Audit Trail would provide added oversight.

❑ DB2 Does Not Provide Adequate Access Security

Although DB2 has some aspects of security contained in it, as discussed above, it is not designed as security software. Its intended purpose is to provide access to data and system services to users of the system. Reliance on DB2 alone for access security is putting the HUDCAPS system at risk of unauthorized access to its data and resources. This situation arises because DB2 controls access to its functions and data through a system of hierarchical levels of authorities. People or groups are assigned to a level of access authority. Each level of access has its own rights and privileges in the system. The problem with this process is the hierarchical nature of granting subsequent access authority.

Individuals who are granted the high level authority such as the System Administrator (SYSADM) authority, can further grant access privileges to specified data and resources to other users. These users in turn can then grant access to these specified resources to additional users. Thus, resource access granting authorities can spread in an uncontrolled manner. Consequently, HUDCAPS data in DB2 are vulnerable to unauthorized, intentional or unintentional modifications. To prevent the uncontrolled spread of access authorizations, CA-Top Secret should replace the DB2 internal security feature now used to control access to HUDCAPS data.

CICS: On-Line Transaction Control Software

CICS is the IBM software that facilitates the development and execution of on-line, interactive applications. CICS provides screen formats, terminal routing and other control information. Under the CICS facility, HUDCAPS operates as a transaction-processing system, where the master file is updated when a complete screen of information is typed at the terminal and transmitted over telecommunications lines. CICS concurrently handles the transmission and processing of these screen transactions entered by users from different terminals. For a particular application such as HUDCAPS, CICS controls all the necessary resources to carry out the requested operation(s), such as file updates. Some examples of resources are data files, software programs, terminals, printers, etc. In short, a resource is any component, hardware or software, that is needed to carry out the purpose of the application. For further discussion of CICS and the mainframe environment see Appendix 2.

❑ Inadequate Restrictions Over Two Powerful CICS Transactions

CICS supports operations called transactions which are usually initiated from terminals. The CICS transactions allow a user to make inquiries about the contents of a data base; update or add to the contents of a data base; or perform calculations, the results of which can be returned to the user's terminal. The CICS software supplies transactions to the user which are unique to CICS operations. These transactions have identification codes (transactions identifiers) that start with the letter "C".

There are several vendor-supplied CICS administrative transactions available which have the ability to control the system and its resources. Two of which, the execution diagnostic facility transaction (CEDF), and the master terminal transaction (CEMT), are especially useful. They are also very powerful, since they have very broad application. They give the user the ability to do many things and use many resources. Consequently, these transactions should be tightly controlled.

We found that the system security administrator for HUD's Hitachi mainframe computer has not maintained sufficient access control over the CEDF and CEMT transactions. Consequently, the HUDCAPS CICS production region and the resources controlled through it are exposed to serious risk of misuse and disruption.

❑ CEDF: Execution Diagnostic Facility Transaction

The Execution Diagnostic Facility (EDF) is a very powerful tool. It runs as a CICS transaction, and is started with the transaction code CEDF. Its normal purpose is to test a software program on-line without having to modify the program or the program preparation procedure. CEDF intercepts execution of the application program at various points and displays information about it at these points. Also displayed are any screens sent by the application program, so that one can communicate with the application program during testing just as a user would on the production system.

CEDF allows programs to be manipulated outside of the CICS production region where the programs are normally run. This situation can expose the program to unauthorized changes which would not be detected, because it is running outside of its normal control environment. Because of its potential for misuse, most installations restrict its use. The external security software such as CA-Top Secret can be used to define the security attributes for the CEDF transaction. Under this procedure, only those authorized to use CEDF can initiate the transaction. The installation can further limit access to resources available to use with CEDF.

We found that over 70 users had the ability to initiate the CEDF transaction in the production region of the computer for HUDCAPS with no justification. Without limiting the access control over the CEDF transaction, HUDCAPS data and software programs are exposed to misuse and disruption caused by intentional or unintentional unauthorized changes.

❑ CEMT: Master Terminal Transaction

The CEMT transaction enables a user to invoke all the master transaction terminal functions. In doing this, the master terminal programs provide dynamic user control of the CICS system. With the CEMT transaction one can make changes and adjustments to the system while it is running. Using this function, a user can inquire and change the values of CICS parameters, change the status of system resources and, if misused, even bring the CICS system down. Furthermore, users of this transaction can control all CICS system operation functions, including the ability to define non-production files to the production region. Other examples of some of the things that can be done through CEMT are controlling the number of tasks, or the number of certain types of tasks, running at any given time, purging a particular task from the system, or enabling or disabling a file to allow controlled access to it by application programs. The CEMT transaction is the most powerful CICS command and it can significantly affect the system and its users. It is critical, therefore, that the CEMT transaction be given adequate security protection especially in a production system.

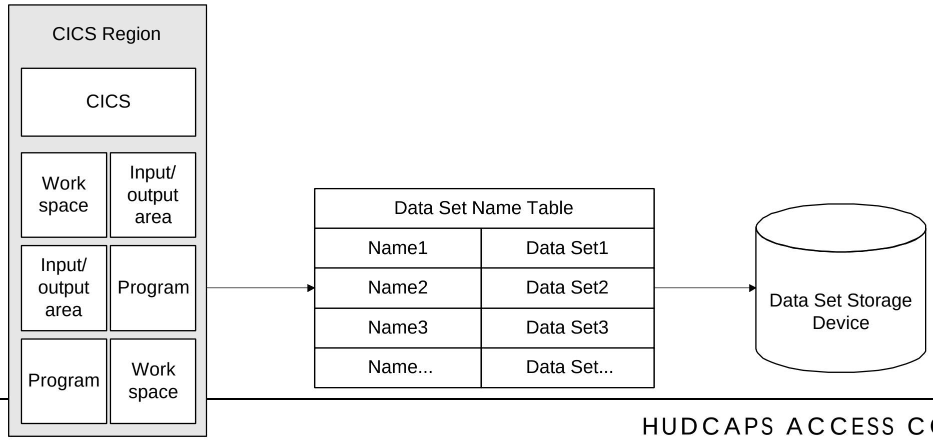
We found that several dozen users had the ability to execute the CEMT transaction in the HUDCAPS CICS production region. We found no reason for these individuals to have the capability to use this powerful and potentially disruptive transaction. Once again, this situation increases the risk of misuse and disruption of computer operations and programs.

❑ Increased Risk Associated With No Dataset Checking and CEMT

Many software programs in HUDCAPS share files or datasets as they are technically known, in carrying out their functions. These programs refer to the shared datasets by their unique names. In order to avoid having to re-code each program with a new name each time that the shared dataset is changed, the CICS system maintains a table of datasets related to the program referenced names. In this way, when a new dataset is needed to be substituted for an old dataset, only the table has to be changed and not the program itself.

Below is an illustration of the relationship of the CICS region and the dataset name table.

Data Set Name Table



Data Set Name Table	
Name1	Data Set1
Name2	Data Set2
Name3	Data Set3
Name...	Data Set...

Insuring that only authorized users have access to these dataset name tables is vital in order to insure that datasets are properly secured. Anyone who can both access and alter these tables and datasets can modify or substitute data without having to have access to the programs using the data. This risk currently exists in the HUD Hitachi mainframe system.

As noted above, several users, have the privilege of no dataset access checking added to their userids. This gives them the ability to access any dataset on the mainframe without any security check. Also, there are several userids who have unrestricted use of the powerful CEMT transaction. With this privilege, they can modify any table such as the dataset name table and disable any audit trail facility. Together, these two abilities could permit a user to manipulate or substitute any data or dataset, which would enable a perpetrator to disrupt or misuse a production program, such as HUDCAPS, without the knowledge of the program users. As an example of the potential harm that could be done, an individual having both of these privileges could alter a dataset containing vendor payee names to his or someone else's benefit without authorization. This action would not be detected by the users of the system.

❑ Underuse of the Resource Definition ON-line (RDO) Feature

CICS needs to know about the characteristics of HUD's mainframe computer system resources so that it can configure the environment and communicate with those system resources. Resources include software assets like programs and data. Hardware resources include terminals, printers and teleprocessing links. The properties of each of these resources vary immensely, yielding a large number of possible configurations.

Resources are defined to CICS in a way that CICS knows which resources to use, what their properties are, and how it can use them. Results from the CICS resources definition process are information held internally by CICS. This information, stored in tables, is used by CICS to control the interaction between resources and interaction between programs and terminals.

CICS operation is based on a variety of control tables that define the characteristics of the different resources. All information regarding the terminals, data files, programs, transactions, and operator identification is contained in these tables. There are some seventeen control tables used in the CICS system. However, there are three which are of particular importance. They are the Program Control Table (PCT), which defines resources for transactions; the Processing Program Table (PPT), which defines resources for programs; and the Terminal Control Table (TCT), which contains resource definitions for all network terminals.

There are three ways that resources can be defined to the CICS tables: (1) by using the Automatic Installation (Autoinstall) feature for dynamic terminal definitions; (2); by using Macros, batch (off-line) definition method; or (3) by using the interactive Resource Definition On-line (RDO) facility. The Autoinstall and the Macro methods are described in Appendix 2.

The new and preferred method of resource definition is the interactive RDO method. RDO is a new CICS software feature that provides the ability to create resource definitions interactively and store them in a single file, the CICS System Definition file (CSD file). RDO is more than just a means of defining resources to the CICS system. It is also a resource management tool. For, beyond the basic capability of interactively defining resources, RDO also permits modifying, deleting and viewing them. It also provides commands for managing groups and lists of definitions. The RDO is an on-line resource definition, inquiry, and installation process.

The RDO facility has many advantages over the use of the older batch/off-line Macro Method to define resources. For one, it is generally easier to use. A series of pre-defined screens are used, which require no programming or compilation of programs. In addition, RDO provides a full audit trail for tracking changes to the system. Since there is only one file to deal with, the CSD file, the process of change control is better managed. Also, since the process is fully automated, there is better accountability over modifying, adding and deleting resources and at a lower expense. Further, the process under RDO is more secure then using the macros method, because there are fewer people involved.

Despite these advantages, we found that the RDO facility was not yet universally used for resource definition. RDO was being used for maintaining IBM mainframe operating system and third party software, but was not used for users' applications such as HUDCAPS. Resources for the users' applications were still being defined with the older off-line Macro Method. HUD systems professionals told us that although they believed in the advantages of RDO, it was more convenient to use the existing Macro Method. However, they also said that all future applications would be configured with RDO.

❑ Inadequate Restriction over CEDA Transaction for RDO

Access to the RDO facility is primarily gained by initiating the CEDA transaction. This transaction permits access to the full set of on-line resource definition commands and, therefore, is very powerful. With CEDA, one can interactively define resources, modify or delete those definitions, check and view them. CEDA also provides commands for managing groups and lists. These include dynamically installing a group of resource definitions on an active system, that is, they take effect immediately.

While we completely support and encourage the use of the RDO facility, because of its many advantages, as described above, we also note that transactions used to implement it are powerful and therefore must be controlled. This is especially true of the CEDA transaction. In this regard, we found that several dozen userids had the ability to issue the CICS CEDA transaction which would enable them to administer the CICS system table in the HUDCAPS production CICS region. As stated above, the CEDA transaction can be used to interactively define and add new programs, transactions, datasets, profiles, terminals, terminal types, connections, and sessions. Because of the critical functions it supports, the CEDA transaction is a very powerful CICS transaction. As a result, adverse unauthorized changes could be made if access to this transaction is not adequately controlled.

Recommendations

Director, Office of Information Technology

(1) Reconfigure the CA-Top Secret security software to operate at global Fail Mode. Also, ensure that all system facilities are in Fail Mode;

(2) Establish a written management policy and practice in which no systems are deployed unless they contain the proper level of security. The required level of security is established by the risk assessment methodology in the System Development Methodology.

(3) Perform a cost/benefit analysis to determine the cost of removing the internal DB2 security from the HUDCAPS/FFS application and replacing it with CA-Top Secret/DB2 security.

(4) Prepare and implement policy to put INSTALL SYSADM under the control of an emergency-only procedure.

(5) Invoke the Audit Trace option. Fine-tune which classes of information will be collected for the HUDCAPS/FFS application.

(6) Change CA-Top Secret so that only DB2 and MVS systems programmers update access to DB2 system datasets.

(7) Assist the HUDCAPS/FFS program staff to establish a formal security administration function to control and administer DB2 security.

(8) Group and assign DB2 secondary IDs according to assigned job functions.

(9) Convert the application definitions to CICS System Definition File (CSD) and manage them using Resource Definition On-line (RDO).

(10) Remove the NODSNCHK security bypass privilege, also expressed as DATASET = *. , ACCESS = ALL, once the dataset high level qualifiers used by the production region are identified and assigned to the HUDCAPS' CICS Region Userid.

(11) Remove the ability for systems programmers and DB2 administrators to submit batch jobs under the authority of the userid assigned to the CICS production region.

(12) Restrict the use of the CEMT transaction to only a few system operators and the system programmers. In addition, use CA-Top Secret to restrict the specific functions of the CEMT command.

(13) Remove CEDA transaction access from all non-CICS system programmers.

(14) Remove CEDF transaction access from all non-CICS system programmers.

Appendix A

Background On CA-Top Secret

CA-Top Secret is an access control security package sold by Computer Associates International, Inc. (CA). It is an hierarchical approach starting from the highest level of the computer environment. When properly implemented, CA-Top Secret is designed to protect against unauthorized access and to permit users to perform only authorized functions. CA-Top Secret controls access to resources by limiting system entry and limiting how, when, and which resources a user can access in the system. CA-Top Secret is now installed on the Department's Hitachi mainframe computer. The HUDCAPS system runs on the Hitachi.

Identifying Users

CA-Top Secret controls access to a system by first identifying users through an Accessor ID (ACID). ACIDs can be assigned to individuals or groups of individuals such as departments or divisions. This technique permits the assignment of ACIDs, if desired, to mirror an organizational structure. When a group of users needs to use a set of identical resources in the same way, it is convenient to define this set of access authorizations once and then associate the entire set with each of the users in the group. In CA-Top Secret, this set of common resource access characteristics is termed a *profile*. Every profile is assigned a unique profile ACID. There are also special ACID types, called Control ACIDs, which are used by the Security Administrator to create, maintain and protect the integrity of the CA-Top Secret security implementation. ACIDs are controlled by a system of password validation.

Securing Resources

A resource is any component of the computing or operating system required by a job or task. Resources include such things as the main storage area of the computer, input and output devices such as disk and tape drives and terminals, central processing units, datasets or files of data, programs, collections or libraries of programs, etc. Resources are secured by first defining them to CA-Top Secret, and then by providing access authorizations for each ACID that will need to use those resources. Authorization is the permission required to access a particular resource. If an ACID does not *own* a resource, the user must be authorized to access it. When a resource is *owned* by a particular ACID, that ACID has unlimited access to that resource. All other ACIDs must be specifically authorized to access the resource.

As part of the defining process, it is necessary to define users as well as resources to CA-Top Secret. Users are defined through their ACIDs. Resources are defined in a two-step process. In the first step, resources are defined by class (terminals, data files, programs, etc.), and in the second step, by the owner of the resource. For example, terminal PD01 would be defined first as a member of the resource class “TERMINAL,” and then as an individual resource owned by ACID USER01.

There are several different ways of authorizing an ACID to have access to a given resource. The first, as stated above, is to designate the ACID as the resource *owner*. The second is to grant specific permission to access that resource. This is done by means of the CA-Top Secret PERMIT command. By adding appropriate keywords to this command, one can further restrict an ACID’s access to a resource. For example, an ACID could be restricted to the source of access (e.g., a designated terminal), the time and date of access, the facility to access, the path of access (e.g., an ACID would be able to access a resource only through a specific program, loaded from a specific library, etc.), the level of access (READ, WRITE, UPDATE, etc.), and the action taken by CA-Top Secret when a particular access request is received.

This last restriction available to the PERMIT command, the action taken by CA-Top Secret when a particular access request is received, provides another layer of security and even finer tunes the ability to restrict or grant access to an ACID. This keyword, called the ACTION keyword, when used with the PERMIT command, can do such things as deny the ACID access to the resource; create an audit trail when the resource is accessed; add additional password protection before granting access to a dataset; tell CA-Top Secret, for requests for access to datasets, to check only volume (i.e., storage device) authorizations for the device on which they reside, instead of authorization checks for both the volume and dataset; etc.

There is also a global access authorization sometimes given to a particular resource. In that case, that resource would be available to all ACIDs without any additional authorization needed.

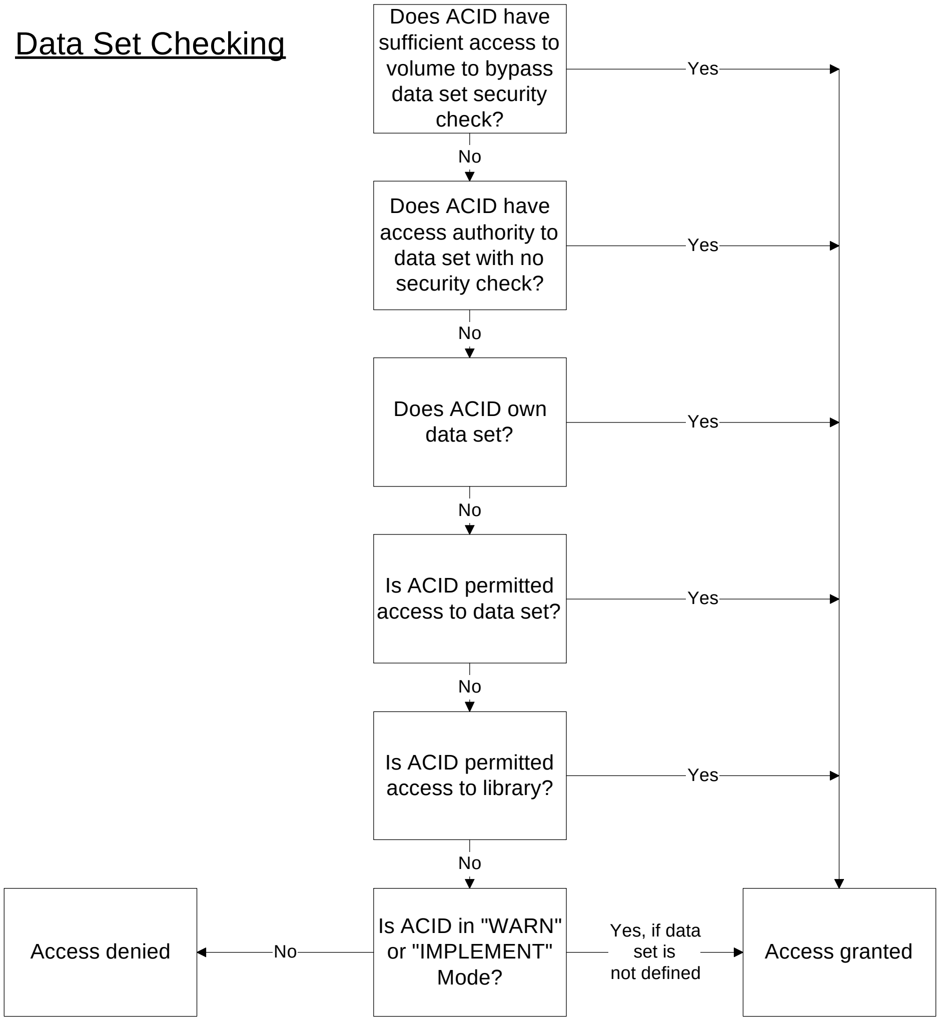
Once all resources have been defined to CA-Top Secret and their access levels have been specified, any future request to access those resources is processed through the CA-Top Secret *Security Validation Algorithm*. This algorithm is the formula CA-Top Secret uses to determine whether an ACID has the appropriate authorization to access a particular resource. There are many factors considered by the *Security Validation Algorithm*. However, there are several key factors. One such factor considered is the ownership of the resource. If the ACID has ownership, the user is given access. Another factor considered in the validation algorithm is the checking of various records for the PERMIT command. Since there may be several PERMIT commands in different records, part of the algorithm’s function is to sort them out and, following its internal rules, select the proper access authorization.

The security *mode* settings are also considered. There are four modes of operation for CA-Top Secret. The mode settings can make the difference between an unauthorized attempt at access failing outright, or being accepted with a warning. This topic is discussed in detail in a separate section below.

In addition to checking for access authorization for the factors discussed above, the *Security Validation Algorithm* also determines access rights to *volume* and *dataset* resources. Volume is the name given by IBM to an external data input and output device, such as a disk drive or a tape drive. A dataset is a collection of information which can be communicated, interpreted, or processed by automatic means. A dataset can be a collection of records, a set of program instructions, or any other group of information which can be represented in a formal manner.

Datasets reside on a volume. Therefore checking for access rights to a dataset is a two step process. When a request for access to a dataset is sent from an ACID, access authorization is first determined for the volume on which it resides. Since both the volume and dataset will have their own access authorization conditions, the *Security Validation Algorithm* will use its internal rules to determine the results of an access request.

The following figure illustrates the process of dataset checking.


Data Set Checking
Does ACID have sufficient access to volume to bypass data set security check?
Yes
No
Does ACID have access authority to data set with no security check?
Yes
No
Does ACID own data set?
Yes
No
Is ACID permitted access to data set?
Yes
No
Is ACID permitted access to library?
Yes
No
Access denied
No
Is ACID in "WARN" or "IMPLEMENT" Mode?
Yes, if data set is not defined
Access granted

Bypassing Resource Checking

Computing resources are secured by first defining them to CA-Top Secret, and then by providing access authorizations for each ACID that will need to use those resources. Authorization is the permission required to access a particular resource.

CA-Top Secret controls access to a system by first identifying users through an ACID. ACIDs can be assigned to individuals or groups of individuals such as departments or divisions. When a group of users needs to use a set of identical resources in the same way, it is convenient to define this set of access authorizations once and then associate the entire set with each of the users in the group. CICS regions themselves are also assigned an ACID. In that way, all programs running in a particular CICS region are further restricted access to resources depending on the access permission granted to the ACID assigned to that region. See the discussion below under CICS for an explanation of a CICS region.

In certain unusual circumstances special so-called *bypass* attributes can be added to the ACID thereby granting it special privileges in accessing resources. With this bypass attribute added, any time a particular ACID makes a certain type of access request, those requests will bypass security checks. The guidance given on the use of this attribute by CA-Top Secret states: “This type of ‘blank check’ for security access is not recommended for ordinary usage but may be considered when setting up disaster recovery procedures (where security needs to be bypassed in a hurry).” The guide goes on to state: “This type of action is not recommended for most users...bypass options (also called ‘no-check attributes’) should be used with great care if at all.”

Some examples of the bypass options are: NORESCHK, which allows the ACID to bypass all resource checking with the exception of datasets and volumes; NOVOLCHK, which allows the ACID to bypass all volume checking; and NODSNCHK, which allows an ACID to bypass dataset checking. The use of the NODSNCHK bypass privilege, also expressed as DATASET = *., ACCESS = ALL, must be especially controlled. With this privilege, a program can read, update, or delete any data stored on the mainframe computer.

CA-Top Secret Modes of Operation

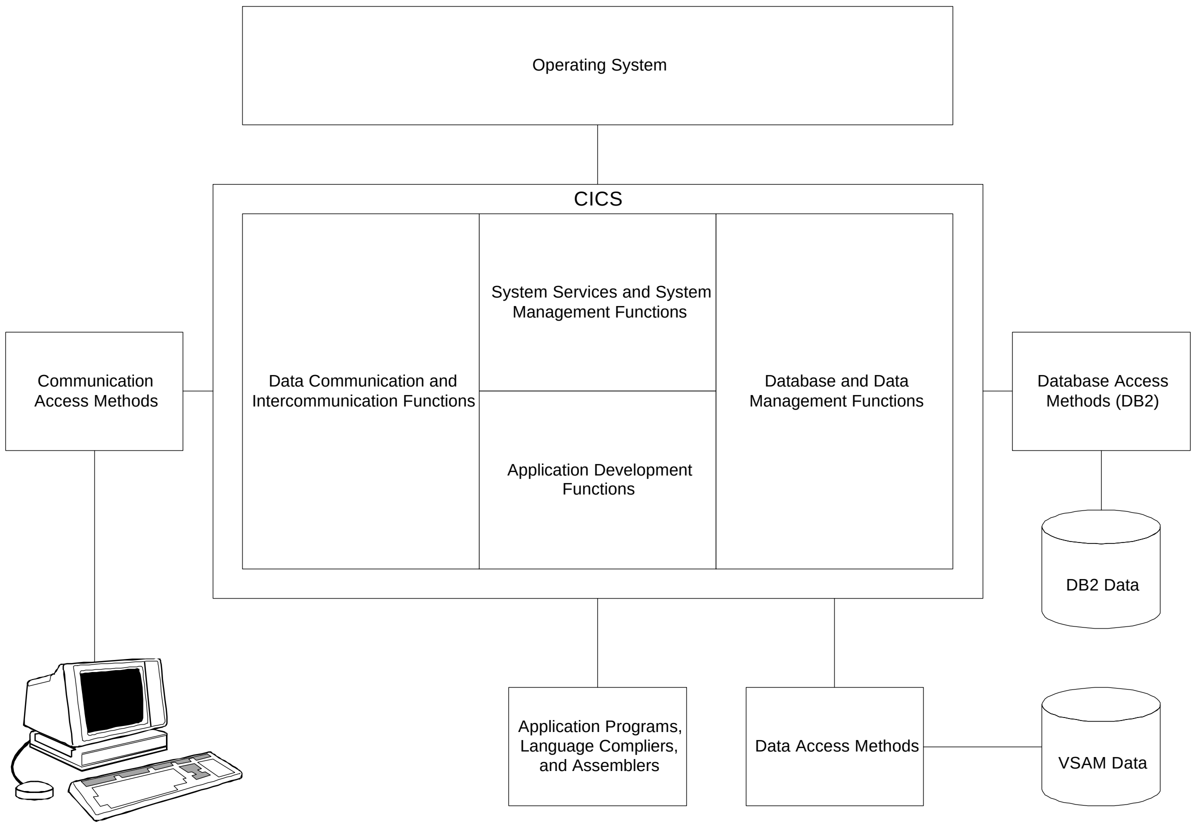
To facilitate its installation, CA-Top Secret has four different modes of operation. Each mode gives increasing control over access to resources. The different modes are used so that an installation can perform a gradual, controlled security implementation, while, at the same time, provide immediate protection to critical resources. Each mode provides a different kind of response to a security violation. Different modes can be set system-wide, by operating system facility, or by type of resource. The four modes are *Dormant*, *Warn*, *Implement*, and *Fail*. CA-Top Secret is initially installed in the Dormant mode. In this mode, CA-Top Secret does not validate user requests, nor normally protect resources. However, critical resources can be protected by exception. This mode is used to control the mechanics of the installation process. The next level of operation, Warn mode, gives administrators an opportunity to adjust the system by examining security reports for undesired results before imposing security restrictions on users. Critical resources, however, are still protected. Violations do not result in failing requests, so processing activity is not affected. Warn mode provides a period of time during which resource utilization can be analyzed, and the system can be monitored for violations. Under Implement mode, CA-Top Secret is active and will fail any unauthorized request from a defined user. Users not defined to CA-Top Secret execute normally, but cannot access protected resources. In Fail mode, CA-Top Secret is in full control of access requests. All users must be defined and all defined resources are protected. Access requests not conforming to an existing access rule will fail.

Appendix B

Background On CICS

Customer Information Control System (CICS) is the IBM software that facilitates the development of on-line, interactive applications. CICS provides screen formats, terminal routing and other control information. Under the CICS facility, HUDCAPS operates as a transaction-processing system, where the master file is updated when a complete screen of information is typed at the terminal and transmitted over telecommunications lines. CICS concurrently handles the transmission and processing of these screen transactions entered by users from different terminals.

CICS Overview

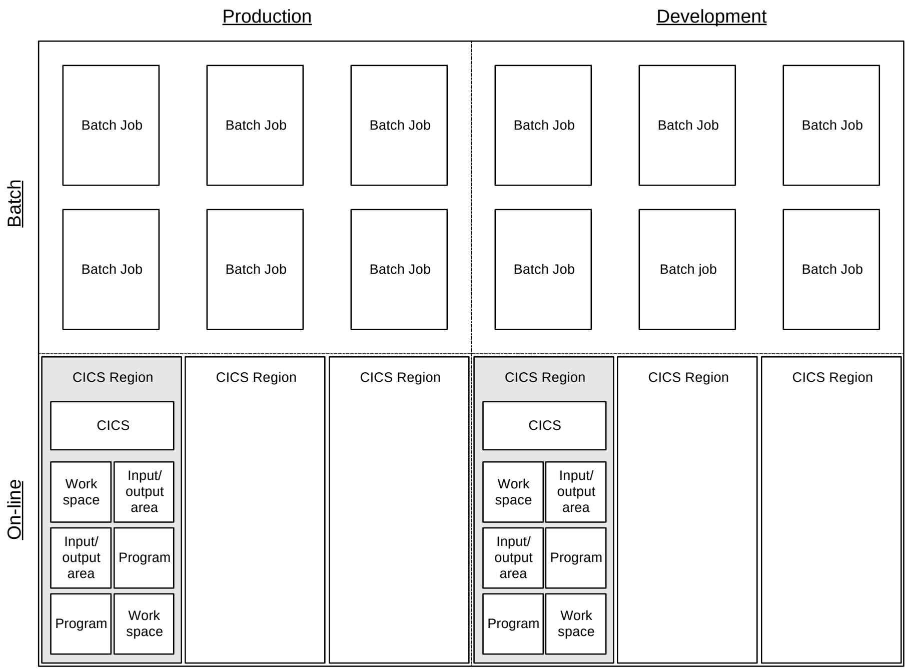

Operating System
CICS
Communication Access Methods
Data Communication and Intercommunication Functions
System Services and System Management Functions
Application Development Functions
Database and Data Management Functions
Database Access Methods (DB2)
DB2 Data
Application Programs, Language Compliers, and Assemblers
Data Access Methods
VSAM Data

The operating environment of the mainframe computer is normally divided, or partitioned, into different logical sections. This is done in order to best control the work performed on the mainframe. For purposes of control and security, all work running on the computer is first divided into two broad categories. One of these two categories contains jobs that are classified as under development. Programs that are still being tested are assigned to this section of the computer. Jobs assigned to this area have not yet been approved to enter into the regular production area of the computer, which is the second major area of operations. Only those programs and applications that have been thoroughly tested and gone through a formal approval process are permitted to enter into the production area. It is critical that these two functions, development and production, be segregated in the operations of the computer in order to safeguard any non-approved software from being run in the production area.

These two major divisions are further subdivided into two other categories of jobs. Jobs running on the mainframe can be categorized depending on the timeliness of their operations and the relative amount of human interaction with the computer. Some jobs are fairly well defined and are not time critical. These are called batch jobs. Batch jobs are viewed by the operating system as being more or less self-contained. No facilities are provided to associate batch jobs to the availability of resources. All such associations must be managed manually by the application programmer. In contrast to the batch job, is the on-line job. On-line jobs are characterized by the need for immediate processing and an interactive environment between the user of the application and the computer.

CICS manages jobs within the on-line region of the computer, thus freeing the programmer from having to be concerned about managing the necessary resources to process his/her job. The on-line section of the operating environment is further subdivided into separate CICS regions. Within these regions, CICS controls all the necessary resources to carry out the needed operations of a particular application like HUDCAPS. See the figure below for a simplified graphical representation of the operating environment.

Operating Environment



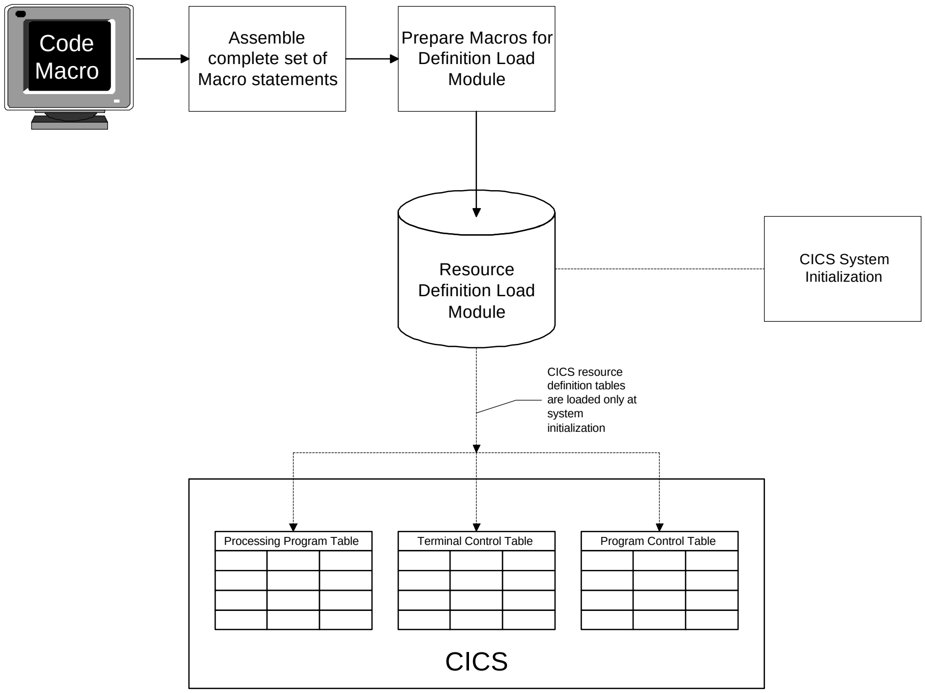
Autoinstall : Resource Definition Method

The Autoinstall feature is used to create resource definitions for certain types of terminals. This method creates resource definitions and installs them in the active CICS system. With Autoinstall, the definition for a terminal is created and installed dynamically in the Terminal Control Table (TCT) at logon time, without the need for the terminal to have its own definition record in the system definition file. The definition is deleted from the TCT when the terminal is logged off. These resource definitions, then, act as models or templates for many terminals of the same type. Each time a terminal requests access to the system, a match is made with a template that fits the terminal type. CICS then installs table entries for these resources dynamically, as and when they are needed.

Macro : Off-line Resource Definition Method

Some resources require that individual definitions be manually coded in a programming language. These programmed definitions are called Macros. The programming language used is called Assembler language. The Macro statements are processed together and put into a file called a Load Module. Each time the CICS system is initialized, the Load Module is used to update the resource definition control tables.

Resource Definition Macro

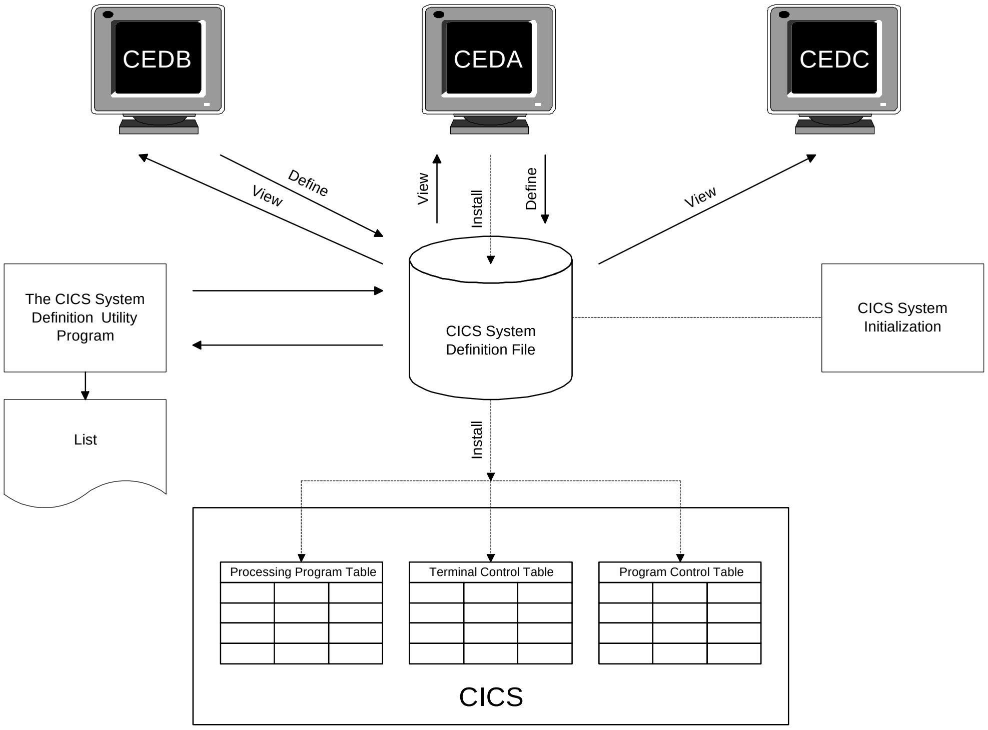

Code Macro
Assemble complete set of Macro statements
Prepare Macros for Definition Load Module
Resource Definition Load Module
CICS System Initialization
CICS resource definition tables are loaded only at system initialization
Processing Program Table
Terminal Control Table
Program Control Table
CICS

RDO: On Line Resource Definition Method

Access to the RDO facility is gained by initiating one of three CICS on-line transactions. These three transactions have the identification codes of CEDA, CEDB, and CEDC. Each transaction perm its access to a different set of RDO definition tools. CEDA permits access to the full set of RDO commands. With CEDA, one can interactively define resources, modify or delete those definitions, and check and view them. CEDA also provides commands for managing groups and lists. These include dynamically installing a group of resource definitions on an active system. That is, they take effect immediately. CEDB allows one to do all the things that CEDA allows except for installing groups of definitions dynamically. Definitions created under the CEDB transaction do not take effect until the beginning of the next regular operating cycle. CEDC is the read or view-only transaction, allowing one to find out which groups are in each list, which resources are defined in each group, and to look at the resource definitions themselves. Using CEDC, one cannot make any changes to the CICS system definition file (CSD) or to the active system.

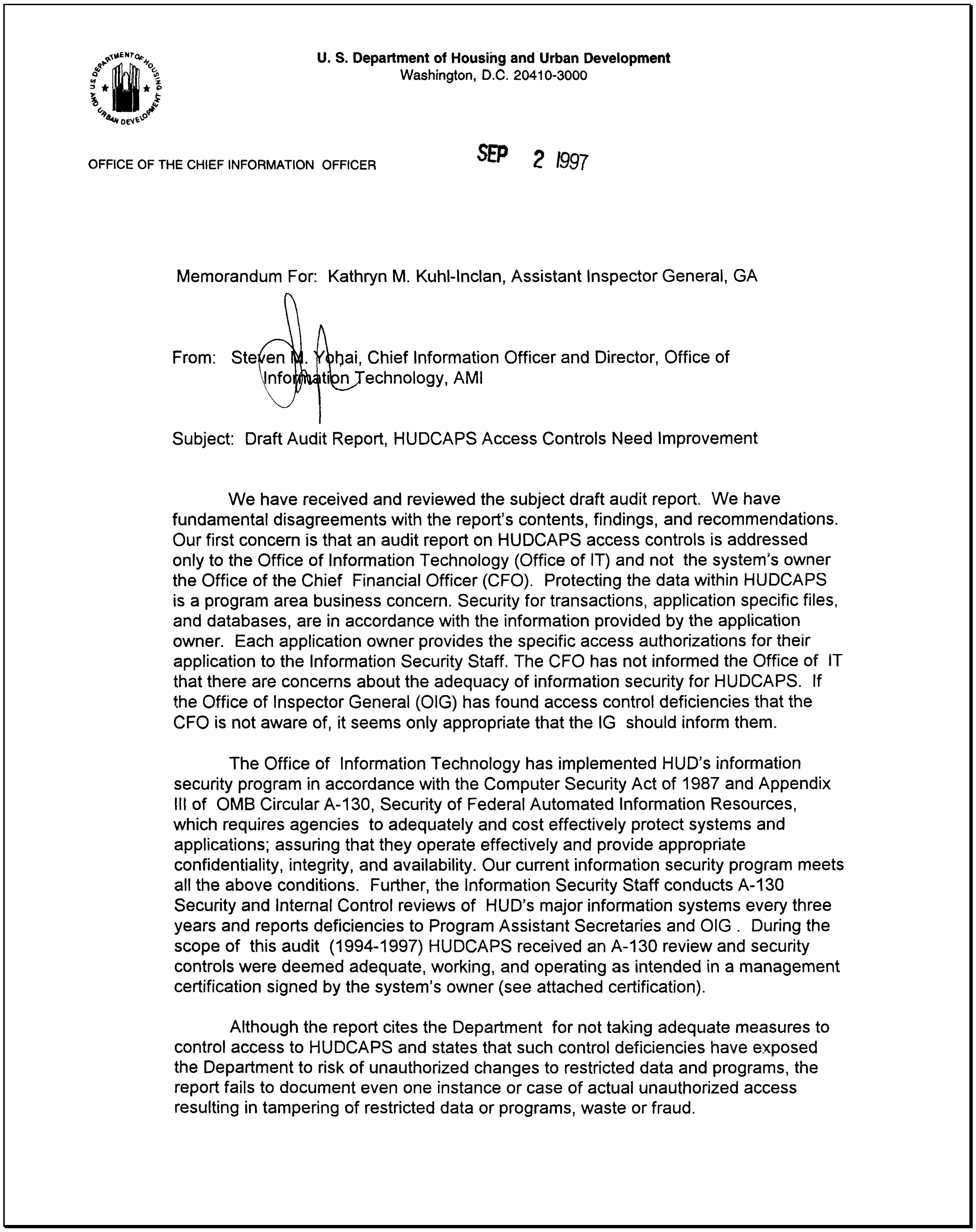
The figure below shows that RDO is an on-line resource definition, inquiry, and installation process. It shows the three transactions interacting with the CSD file. The View command shows the contents of a record on the CSD. The Define command affects the CSD. The Install command is the only one that affects the active CICS system. Install dynamically adds the resource definitions in a group to the active CICS system. A listing of the CSD contents can be gotten with the use of the off-line utility.

Resource Definition Online (RDO) Transactions



Appendix C

Comments From Office of Information Technology



U. S. Department of Housing and Urban Development
Washington, D.C. 20410-3000

OFFICE OF THE CHIEF INFORMATION OFFICER

SEP 2 1997

Memorandum For: Kathryn M. Kuhl-Inclan, Assistant Inspector General, GA

From: Steven M. Yohai, Chief Information Officer and Director, Office of Information Technology, AMI

Subject: Draft Audit Report, HUDCAPS Access Controls Need Improvement

See Comment 1

We have received and reviewed the subject draft audit report. We have fundamental disagreements with the report's contents, findings, and recommendations. Our first concern is that an audit report on HUDCAPS access controls is addressed only to the Office of Information Technology (Office of IT) and not the system's owner the Office of the Chief Financial Officer (CFO). Protecting the data within HUDCAPS is a program area business concern. Security for transactions, application specific files, and databases, are in accordance with the information provided by the application owner. Each application owner provides the specific access authorizations for their application to the Information Security Staff. The CFO has not informed the Office of IT that there are concerns about the adequacy of information security for HUDCAPS. If the Office of Inspector General (OIG) has found access control deficiencies that the CFO is not aware of, it seems only appropriate that the IG should inform them.

See Comment 2

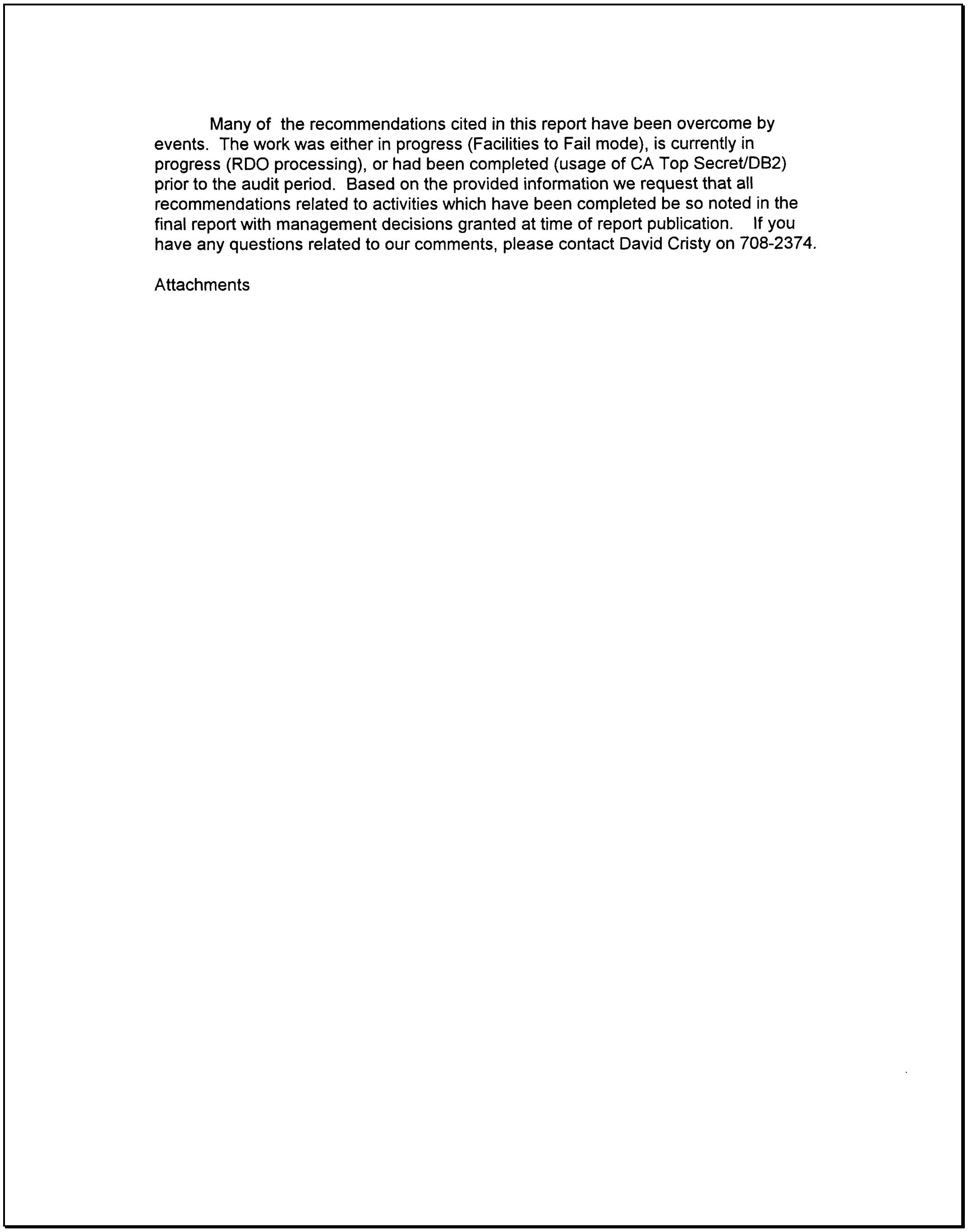
The Office of Information Technology has implemented HUD's information security program in accordance with the Computer Security Act of 1987 and Appendix III of OMB Circular A-130, Security of Federal Automated Information Resources, which requires agencies to adequately and cost effectively protect systems and applications; assuring that they operate effectively and provide appropriate confidentiality, integrity, and availability. Our current information security program meets all the above conditions. Further, the Information Security Staff conducts A-130 Security and Internal Control reviews of HUD's major information systems every three years and reports deficiencies to Program Assistant Secretaries and OIG . During the scope of this audit (1994-1997) HUDCAPS received an A-130 review and security controls were deemed adequate, working, and operating as intended in a management certification signed by the system's owner (see attached certification).

See Comment 3

Although the report cites the Department for not taking adequate measures to control access to HUDCAPS and states that such control deficiencies have exposed the Department to risk of unauthorized changes to restricted data and programs, the report fails to document even one instance or case of actual unauthorized access resulting in tampering of restricted data or programs, waste or fraud.

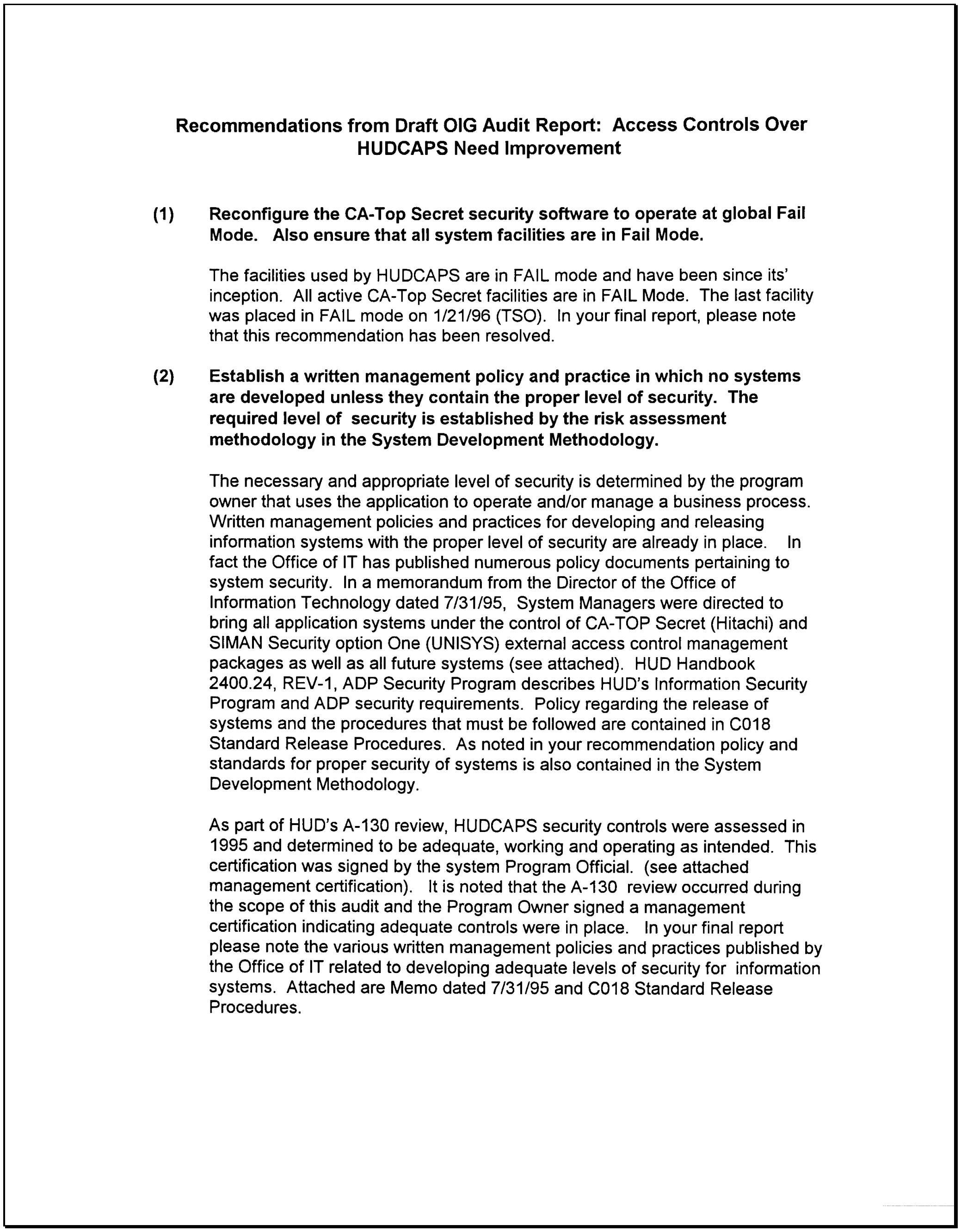
See Comment 4

See
Comment 5



Many of the recommendations cited in this report have been overcome by events. The work was either in progress (Facilities to Fail mode), is currently in progress (RDO processing), or had been completed (usage of CA Top Secret/DB2) prior to the audit period. Based on the provided information we request that all recommendations related to activities which have been completed be so noted in the final report with management decisions granted at time of report publication. If you have any questions related to our comments, please contact David Cristy on 708-2374.

Attachments



Recommendations from Draft OIG Audit Report: Access Controls Over HUDCAPS Need Improvement

(1) Reconfigure the CA-Top Secret security software to operate at global Fail Mode. Also ensure that all system facilities are in Fail Mode.

See Comment 6

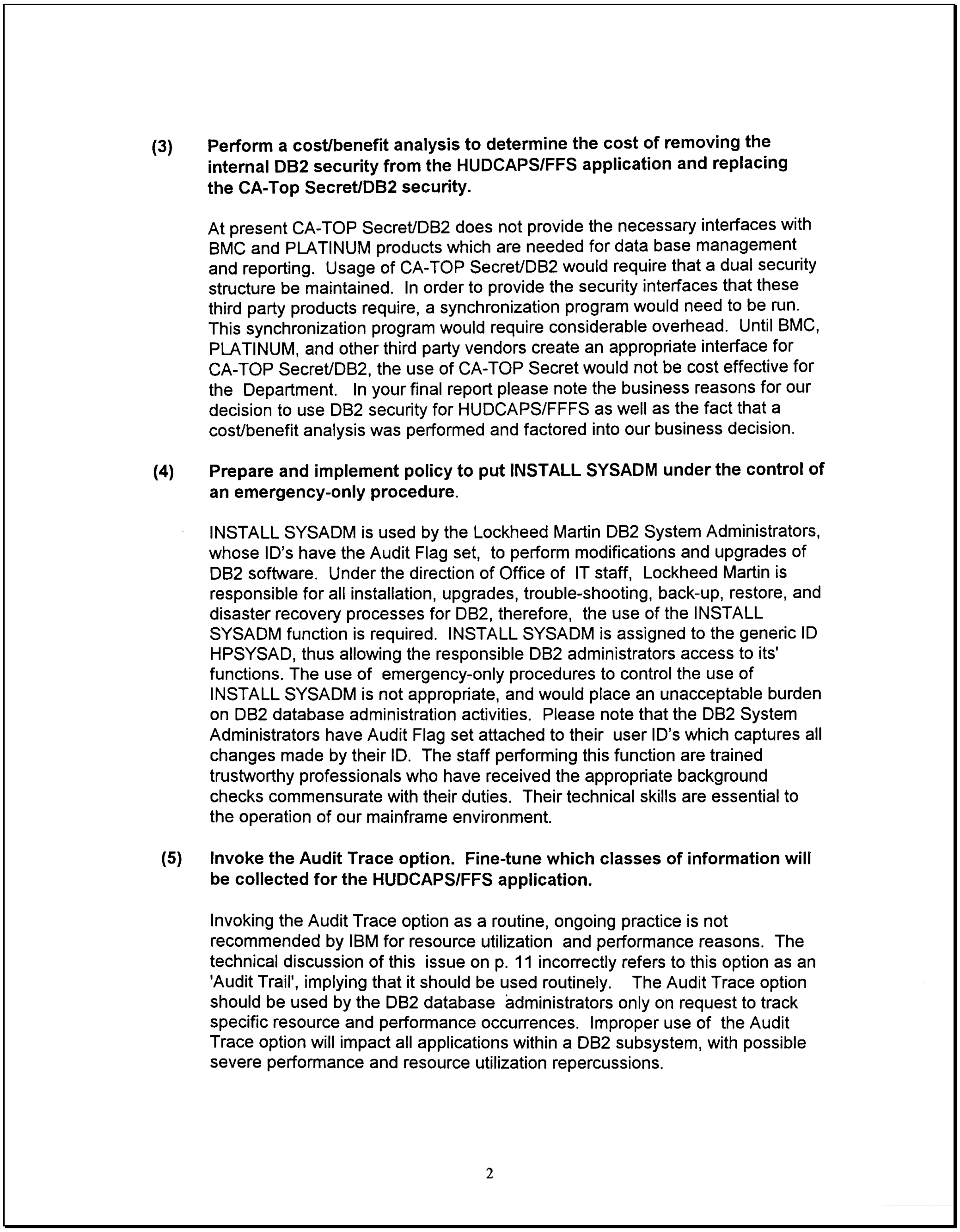
The facilities used by HUDCAPS are in FAIL mode and have been since its' inception. All active CA-Top Secret facilities are in FAIL Mode. The last facility was placed in FAIL mode on 1/21/96 (TSO). In your final report, please note that this recommendation has been resolved.

(2) Establish a written management policy and practice in which no systems are developed unless they contain the proper level of security. The required level of security is established by the risk assessment methodology in the System Development Methodology.

See Comment 7

The necessary and appropriate level of security is determined by the program owner that uses the application to operate and/or manage a business process. Written management policies and practices for developing and releasing information systems with the proper level of security are already in place. In fact the Office of IT has published numerous policy documents pertaining to system security. In a memorandum from the Director of the Office of Information Technology dated 7/31/95, System Managers were directed to bring all application systems under the control of CA-TOP Secret (Hitachi) and SIMAN Security option One (UNISYS) external access control management packages as well as all future systems (see attached). HUD Handbook 2400.24, REV-1, ADP Security Program describes HUD's Information Security Program and ADP security requirements. Policy regarding the release of systems and the procedures that must be followed are contained in C018 Standard Release Procedures. As noted in your recommendation policy and standards for proper security of systems is also contained in the System Development Methodology.

As part of HUD's A-130 review, HUDCAPS security controls were assessed in 1995 and determined to be adequate, working and operating as intended. This certification was signed by the system Program Official. (see attached management certification). It is noted that the A-130 review occurred during the scope of this audit and the Program Owner signed a management certification indicating adequate controls were in place. In your final report please note the various written management policies and practices published by the Office of IT related to developing adequate levels of security for information systems. Attached are Memo dated 7/31/95 and C018 Standard Release Procedures.



(3) Perform a cost/benefit analysis to determine the cost of removing the internal DB2 security from the HUDCAPS/FFS application and replacing the CA-Top Secret/DB2 security.

See
Comment 8

At present CA-TOP Secret/DB2 does not provide the necessary interfaces with BMC and PLATINUM products which are needed for data base management and reporting. Usage of CA-TOP Secret/DB2 would require that a dual security structure be maintained. In order to provide the security interfaces that these third party products require, a synchronization program would need to be run. This synchronization program would require considerable overhead. Until BMC, PLATINUM, and other third party vendors create an appropriate interface for CA-TOP Secret/DB2, the use of CA-TOP Secret would not be cost effective for the Department. In your final report please note the business reasons for our decision to use DB2 security for HUDCAPS/FFFS as well as the fact that a cost/benefit analysis was performed and factored into our business decision.

(4) Prepare and implement policy to put INSTALL SYSADM under the control of an emergency-only procedure.

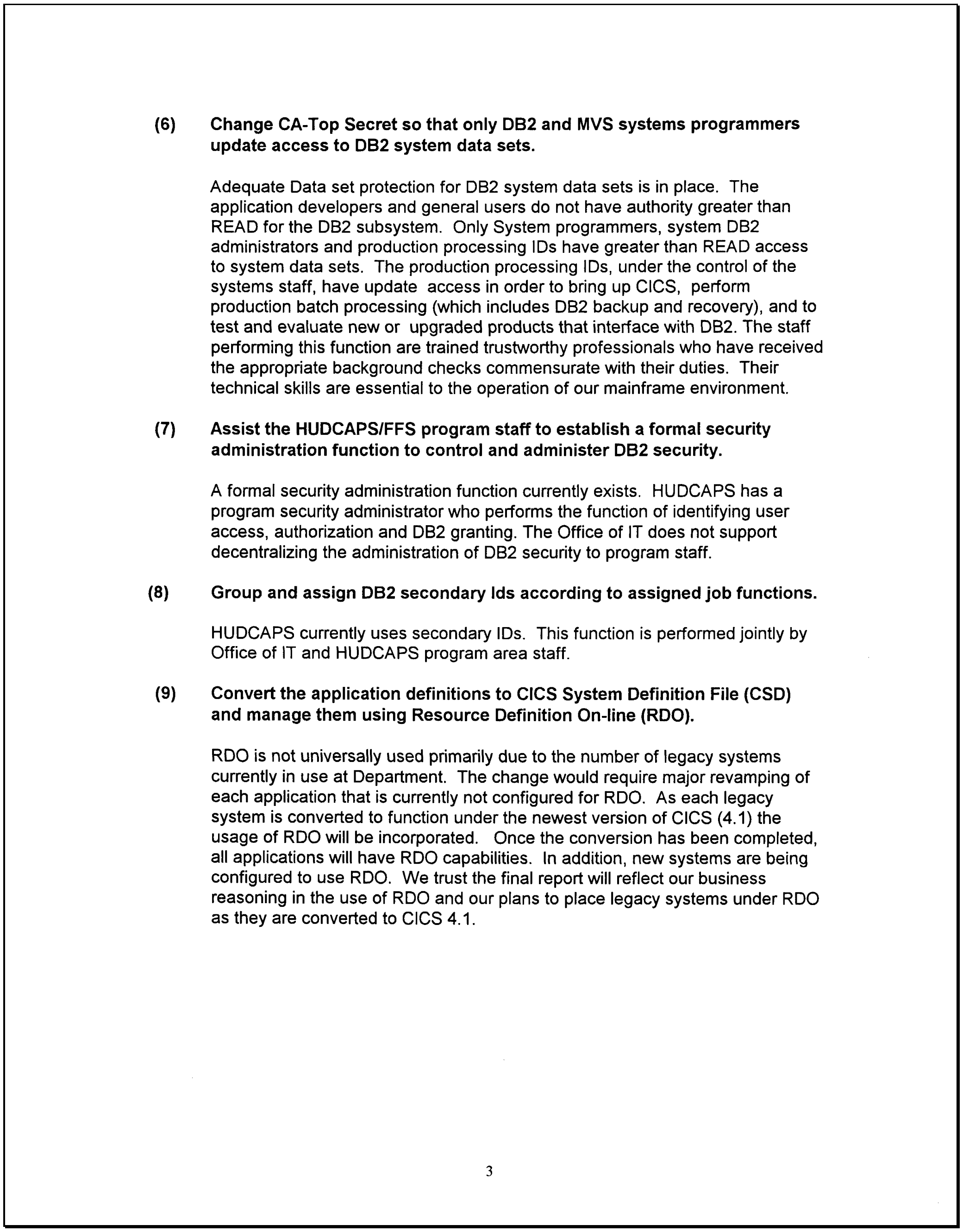
See
Comment 9

INSTALL SYSADM is used by the Lockheed Martin DB2 System Administrators, whose ID's have the Audit Flag set, to perform modifications and upgrades of DB2 software. Under the direction of Office of IT staff, Lockheed Martin is responsible for all installation, upgrades, trouble-shooting, back-up, restore, and disaster recovery processes for DB2, therefore, the use of the INSTALL SYSADM function is required. INSTALL SYSADM is assigned to the generic ID HPSYSAD, thus allowing the responsible DB2 administrators access to its' functions. The use of emergency-only procedures to control the use of INSTALL SYSADM is not appropriate, and would place an unacceptable burden on DB2 database administration activities. Please note that the DB2 System Administrators have Audit Flag set attached to their user ID's which captures all changes made by their ID. The staff performing this function are trained trustworthy professionals who have received the appropriate background checks commensurate with their duties. Their technical skills are essential to the operation of our mainframe environment.

(5) Invoke the Audit Trace option. Fine-tune which classes of information will be collected for the HUDCAPS/FFS application.

See
Comment 10

Invoking the Audit Trace option as a routine, ongoing practice is not recommended by IBM for resource utilization and performance reasons. The technical discussion of this issue on p. 11 incorrectly refers to this option as an 'Audit Trail', implying that it should be used routinely. The Audit Trace option should be used by the DB2 database administrators only on request to track specific resource and performance occurrences. Improper use of the Audit Trace option will impact all applications within a DB2 subsystem, with possible severe performance and resource utilization repercussions.



(6) Change CA-Top Secret so that only DB2 and MVS systems programmers update access to DB2 system data sets.

See Comment 11

Adequate Data set protection for DB2 system data sets is in place. The application developers and general users do not have authority greater than READ for the DB2 subsystem. Only System programmers, system DB2 administrators and production processing IDs have greater than READ access to system data sets. The production processing IDs, under the control of the systems staff, have update access in order to bring up CICS, perform production batch processing (which includes DB2 backup and recovery), and to test and evaluate new or upgraded products that interface with DB2. The staff performing this function are trained trustworthy professionals who have received the appropriate background checks commensurate with their duties. Their technical skills are essential to the operation of our mainframe environment.

(7) Assist the HUDCAPS/FFS program staff to establish a formal security administration function to control and administer DB2 security.

See Comment 12

A formal security administration function currently exists. HUDCAPS has a program security administrator who performs the function of identifying user access, authorization and DB2 granting. The Office of IT does not support decentralizing the administration of DB2 security to program staff.

(8) Group and assign DB2 secondary Ids according to assigned job functions.

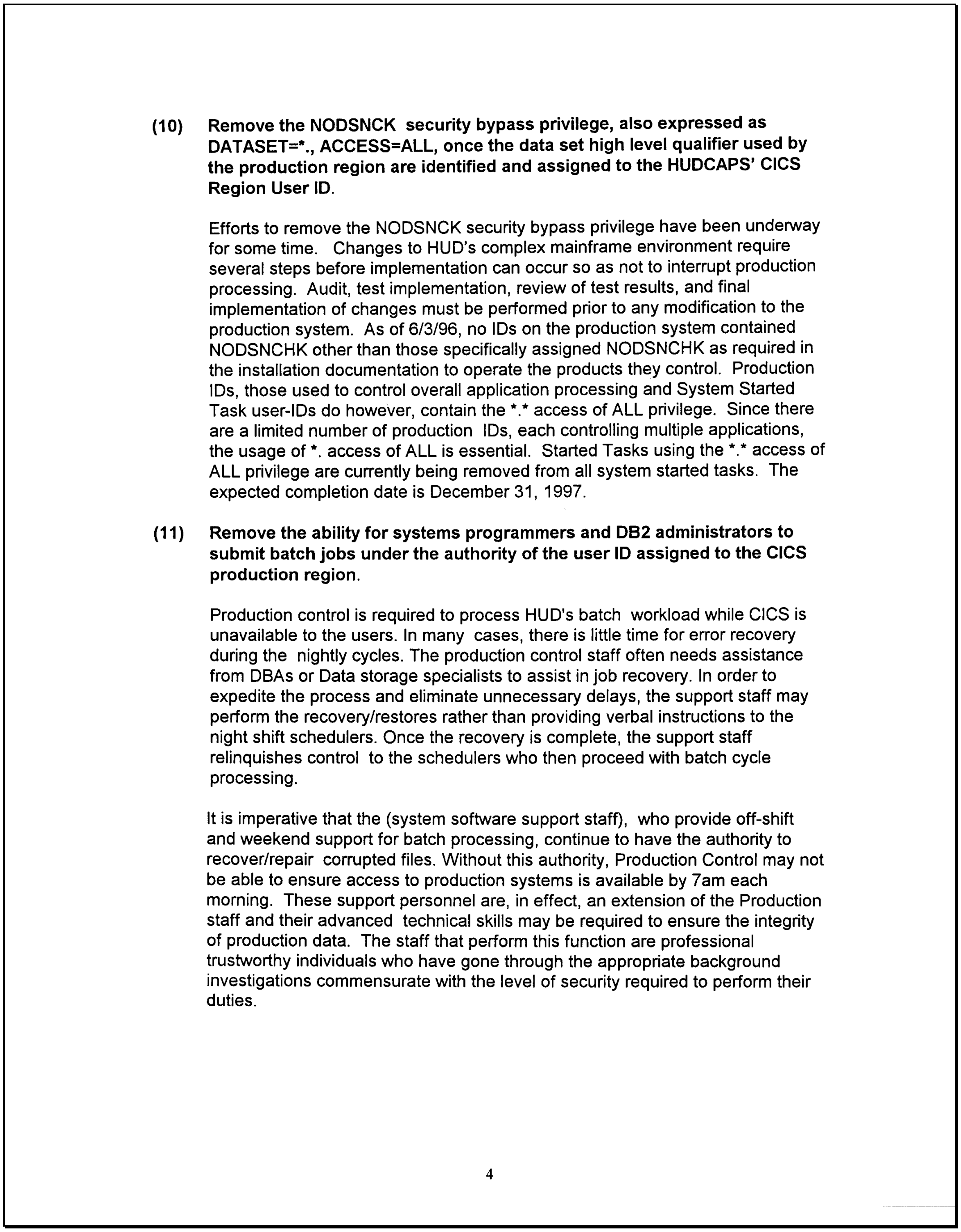
See Comment 13

HUDCAPS currently uses secondary IDs. This function is performed jointly by Office of IT and HUDCAPS program area staff.

(9) Convert the application definitions to CICS System Definition File (CSD) and manage them using Resource Definition On-line (RDO).

See Comment 14

RDO is not universally used primarily due to the number of legacy systems currently in use at Department. The change would require major revamping of each application that is currently not configured for RDO. As each legacy system is converted to function under the newest version of CICS (4.1) the usage of RDO will be incorporated. Once the conversion has been completed, all applications will have RDO capabilities. In addition, new systems are being configured to use RDO. We trust the final report will reflect our business reasoning in the use of RDO and our plans to place legacy systems under RDO as they are converted to CICS 4.1.



(10) **Remove the NODSNCK security bypass privilege, also expressed as DATASET=*., ACCESS=ALL, once the data set high level qualifier used by the production region are identified and assigned to the HUDCAPS' CICS Region User ID.**

See Comment 15

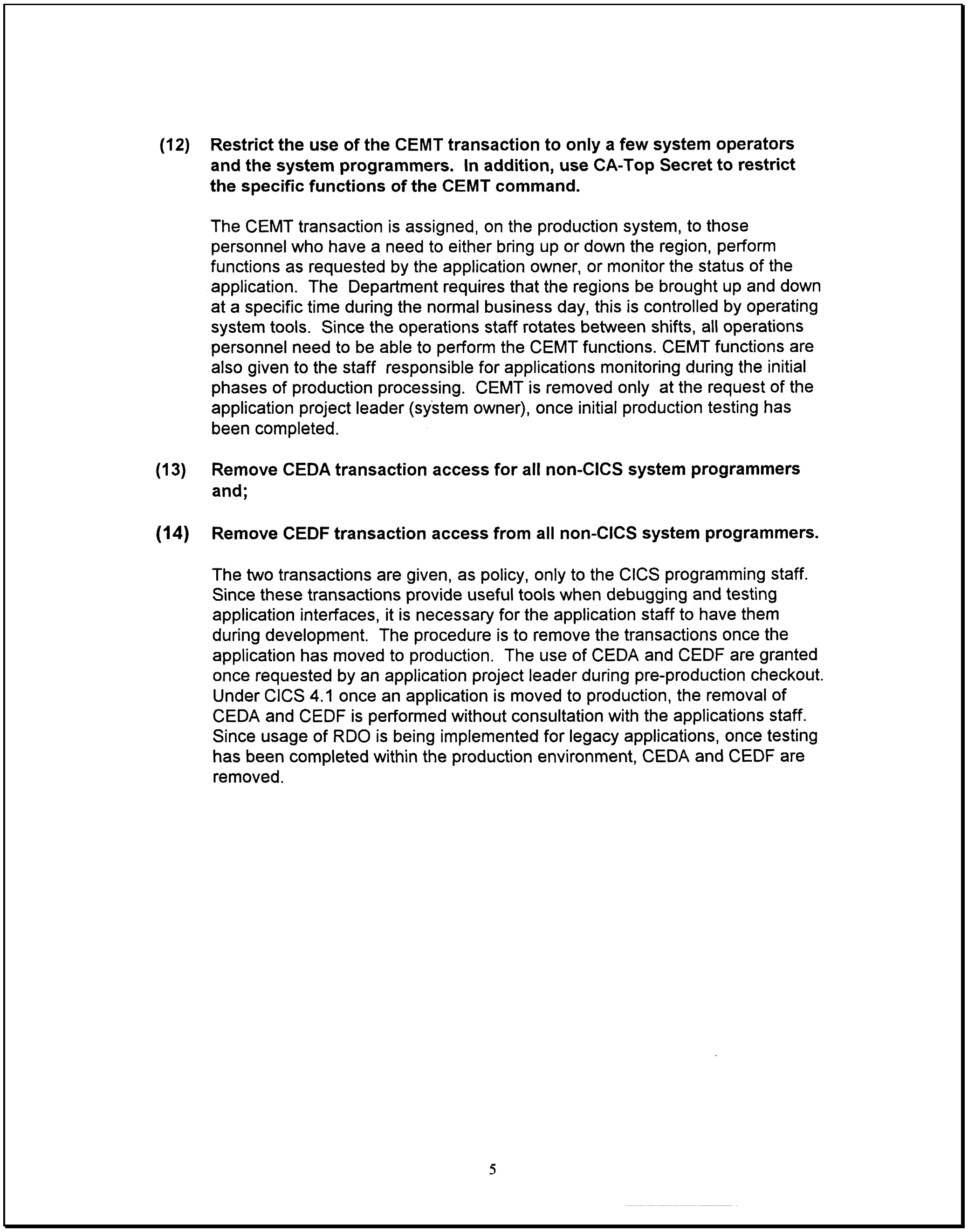
Efforts to remove the NODSNCK security bypass privilege have been underway for some time. Changes to HUD's complex mainframe environment require several steps before implementation can occur so as not to interrupt production processing. Audit, test implementation, review of test results, and final implementation of changes must be performed prior to any modification to the production system. As of 6/3/96, no IDs on the production system contained NODSNCHK other than those specifically assigned NODSNCHK as required in the installation documentation to operate the products they control. Production IDs, those used to control overall application processing and System Started Task user-IDs do however, contain the *.* access of ALL privilege. Since there are a limited number of production IDs, each controlling multiple applications, the usage of *. access of ALL is essential. Started Tasks using the *.* access of ALL privilege are currently being removed from all system started tasks. The expected completion date is December 31, 1997.

(11) **Remove the ability for systems programmers and DB2 administrators to submit batch jobs under the authority of the user ID assigned to the CICS production region.**

See Comment 16

Production control is required to process HUD's batch workload while CICS is unavailable to the users. In many cases, there is little time for error recovery during the nightly cycles. The production control staff often needs assistance from DBAs or Data storage specialists to assist in job recovery. In order to expedite the process and eliminate unnecessary delays, the support staff may perform the recovery/restores rather than providing verbal instructions to the night shift schedulers. Once the recovery is complete, the support staff relinquishes control to the schedulers who then proceed with batch cycle processing.

It is imperative that the (system software support staff), who provide off-shift and weekend support for batch processing, continue to have the authority to recover/repair corrupted files. Without this authority, Production Control may not be able to ensure access to production systems is available by 7am each morning. These support personnel are, in effect, an extension of the Production staff and their advanced technical skills may be required to ensure the integrity of production data. The staff that perform this function are professional trustworthy individuals who have gone through the appropriate background investigations commensurate with the level of security required to perform their duties.



(12) Restrict the use of the CEMT transaction to only a few system operators and the system programmers. In addition, use CA-Top Secret to restrict the specific functions of the CEMT command.

See Comment 17

The CEMT transaction is assigned, on the production system, to those personnel who have a need to either bring up or down the region, perform functions as requested by the application owner, or monitor the status of the application. The Department requires that the regions be brought up and down at a specific time during the normal business day, this is controlled by operating system tools. Since the operations staff rotates between shifts, all operations personnel need to be able to perform the CEMT functions. CEMT functions are also given to the staff responsible for applications monitoring during the initial phases of production processing. CEMT is removed only at the request of the application project leader (system owner), once initial production testing has been completed.

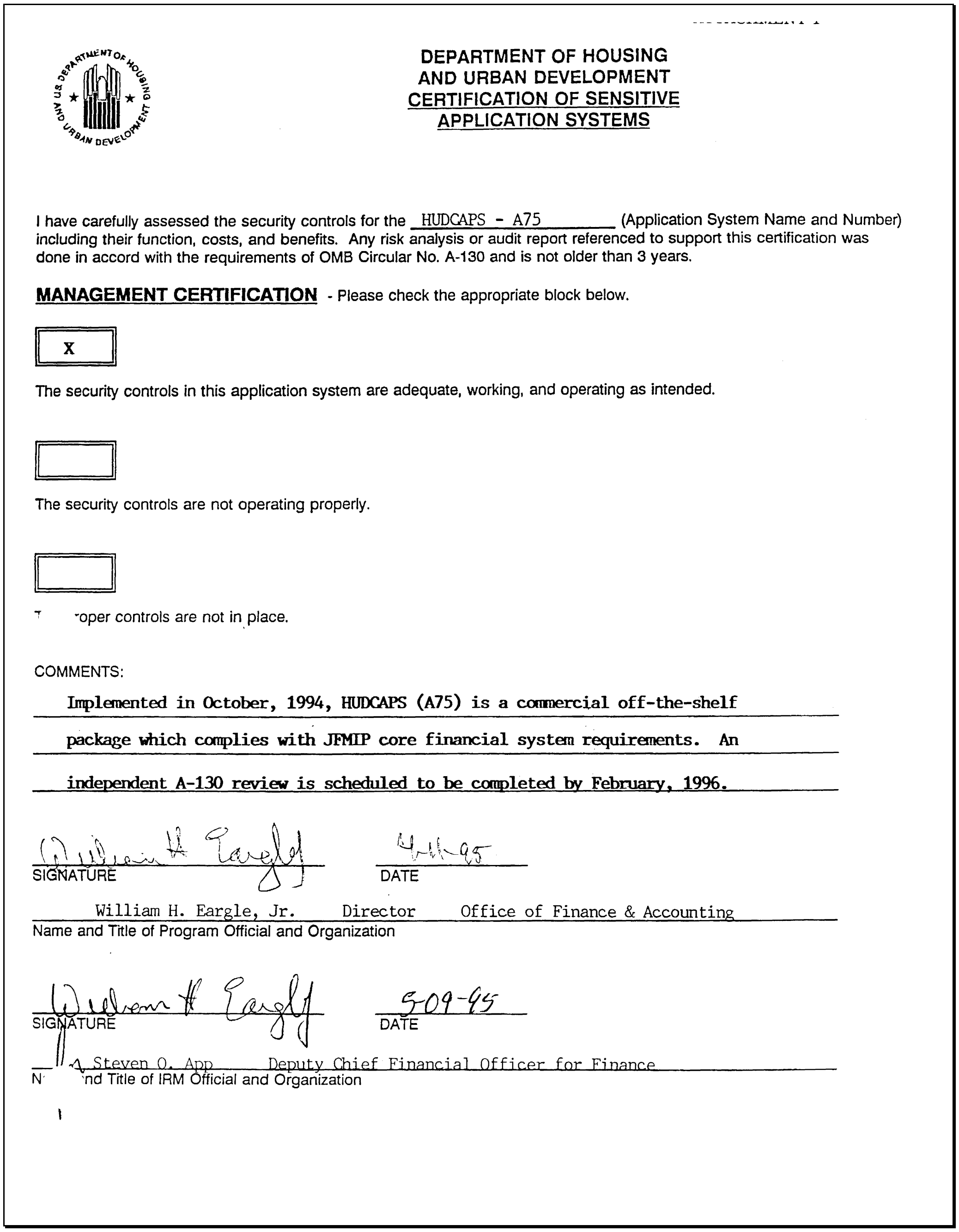
See Comment 18

(13) Remove CEDA transaction access for all non-CICS system programmers and;

(14) Remove CEDF transaction access from all non-CICS system programmers.

See Comment 18

The two transactions are given, as policy, only to the CICS programming staff. Since these transactions provide useful tools when debugging and testing application interfaces, it is necessary for the application staff to have them during development. The procedure is to remove the transactions once the application has moved to production. The use of CEDA and CEDF are granted once requested by an application project leader during pre-production checkout. Under CICS 4.1 once an application is moved to production, the removal of CEDA and CEDF is performed without consultation with the applications staff. Since usage of RDO is being implemented for legacy applications, once testing has been completed within the production environment, CEDA and CEDF are removed.



DEPARTMENT OF HOUSING AND URBAN DEVELOPMENT CERTIFICATION OF SENSITIVE APPLICATION SYSTEMS

I have carefully assessed the security controls for the HUDCAPS - A75 (Application System Name and Number) including their function, costs, and benefits. Any risk analysis or audit report referenced to support this certification was done in accord with the requirements of OMB Circular No. A-130 and is not older than 3 years.

MANAGEMENT CERTIFICATION - Please check the appropriate block below.

☒

The security controls in this application system are adequate, working, and operating as intended.

□

The security controls are not operating properly.

□

oper controls are not in place.

COMMENTS:

Implemented in October, 1994, HUDCAPS (A75) is a commercial off-the-shelf package which complies with JFMIP core financial system requirements. An independent A-130 review is scheduled to be completed by February, 1996.

William H Eargle
SIGNATURE

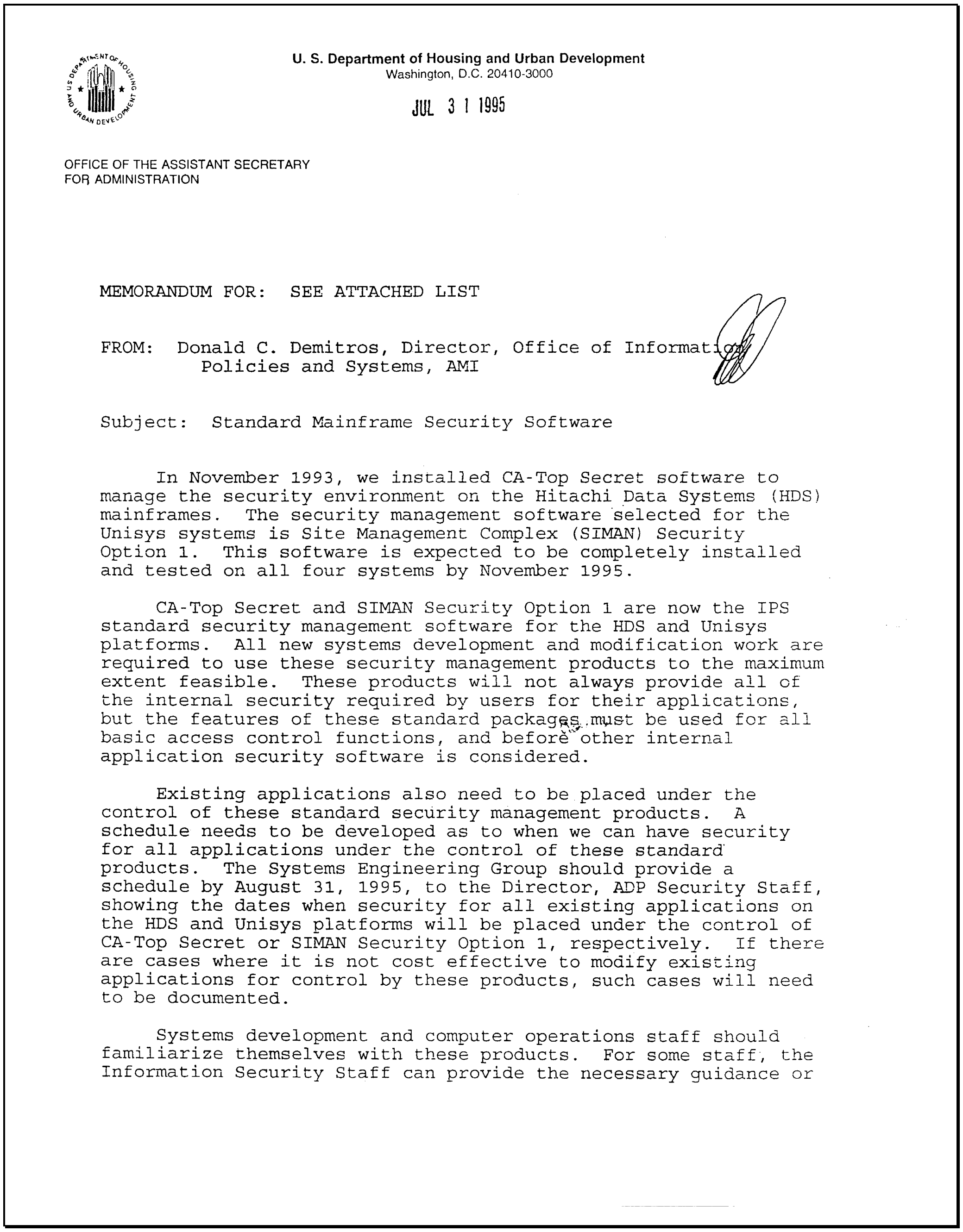
4-11-95
DATE

William H. Eargle, Jr. Director Office of Finance & Accounting
Name and Title of Program Official and Organization

William H Eargle
SIGNATURE

5-09-95
DATE

Steven O. App Deputy Chief Financial Officer for Finance
N nd Title of IRM Official and Organization



U. S. Department of Housing and Urban Development
Washington, D.C. 20410-3000

JUL 3 1 1995

OFFICE OF THE ASSISTANT SECRETARY
FOR ADMINISTRATION

MEMORANDUM FOR: SEE ATTACHED LIST

FROM: Donald C. Demitros, Director, Office of Information
Policies and Systems, AMI

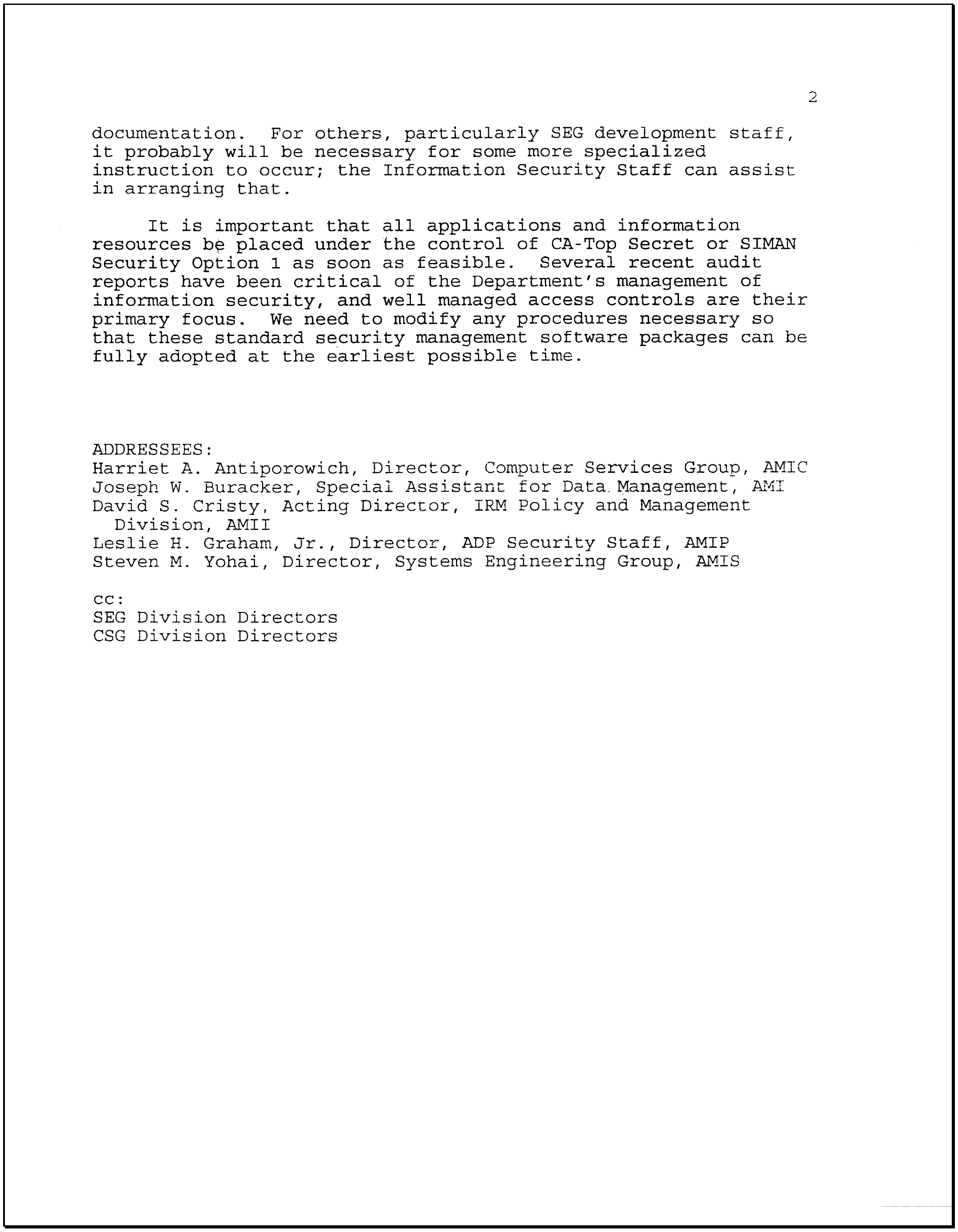
Subject: Standard Mainframe Security Software

In November 1993, we installed CA-Top Secret software to manage the security environment on the Hitachi Data Systems (HDS) mainframes. The security management software selected for the Unisys systems is Site Management Complex (SIMAN) Security Option 1. This software is expected to be completely installed and tested on all four systems by November 1995.

CA-Top Secret and SIMAN Security Option 1 are now the IPS standard security management software for the HDS and Unisys platforms. All new systems development and modification work are required to use these security management products to the maximum extent feasible. These products will not always provide all of the internal security required by users for their applications, but the features of these standard packages must be used for all basic access control functions, and before other internal application security software is considered.

Existing applications also need to be placed under the control of these standard security management products. A schedule needs to be developed as to when we can have security for all applications under the control of these standard products. The Systems Engineering Group should provide a schedule by August 31, 1995, to the Director, ADP Security Staff, showing the dates when security for all existing applications on the HDS and Unisys platforms will be placed under the control of CA-Top Secret or SIMAN Security Option 1, respectively. If there are cases where it is not cost effective to modify existing applications for control by these products, such cases will need to be documented.

Systems development and computer operations staff should familiarize themselves with these products. For some staff, the Information Security Staff can provide the necessary guidance or



documentation. For others, particularly SEG development staff, it probably will be necessary for some more specialized instruction to occur; the Information Security Staff can assist in arranging that.

It is important that all applications and information resources be placed under the control of CA-Top Secret or SIMAN Security Option 1 as soon as feasible. Several recent audit reports have been critical of the Department's management of information security, and well managed access controls are their primary focus. We need to modify any procedures necessary so that these standard security management software packages can be fully adopted at the earliest possible time.

ADDRESSEES:
Harriet A. Antiporowich, Director, Computer Services Group, AMIC
Joseph W. Buracker, Special Assistant for Data Management, AMI
David S. Cristy, Acting Director, IRM Policy and Management
  Division, AMII
Leslie H. Graham, Jr., Director, ADP Security Staff, AMIP
Steven M. Yohai, Director, Systems Engineering Group, AMIS

cc:
SEG Division Directors
CSG Division Directors

IG Comments

1. Even though you 'have fundamental disagreements with our report', we are encouraged that some of your recent activities mentioned in your response appear to address the issues raised in our report. You indicated that you have either already implemented or plan to implement some of our recommendations, for example:

- facilities such as TSO have been placed in FAIL mode in early 1996;
- the NODSNCHK and other security bypass privileges have been removed from all but a few system userids as of mid-1996;
- the 'alternate' security bypass privileges are scheduled to be removed from all system started tasks by the end of 1997;
- the on-line resource definition (RDO) method is being implemented for new systems during installation and for legacy systems during conversion.

We also endorse your action, during the course of our audit, to remove one user ID with all the bypass privileges.

2. We agree that application owners should authorize specific access to their applications based on an employee's job function. However, where identification of application-specific controls is the responsibility of CFO, it is IT's responsibility to identify and to implement general system controls. These controls include appropriate implementation and administration of system software such as CA-Top Secret, CICS, and DB2. Since HUDCAPS depends on these system software for security and processing of data, access to system software must be limited to a controlled number of individuals based on their assigned functions.

3. The A-130 HUDCAPS reviews focus on input, output and processing controls of an application, not on system software supporting HUDCAPS. Without control over system software such as CA-Top Secret, CICS and DB2, HUDCAPS is vulnerable to accidental or intentional unauthorized use.

4. When audit trails are disabled and when security bypass privileges are granted, user activities are not recorded. Without audit trails, neither OIG nor IT will ever know whether an unauthorized access transpired. However, we did notice that since the start of our audit, some of the bypass privileges given to users have been removed.

5. Our continued monitoring of HUDCAPS indicates that most of the control issues we raised in our report still apply. We do not consider it reasonable that over 40 contractor staff have update access to all HUDCAPS data on the Hitachi production system; that over 70 contractor staff have CICS most powerful commands, which can shut down HUDCAPS and/or update any file; that the audit trail for DB2 (database software) has been turned off. Consequently, at this time, we cannot agree to management decisions on all of the recommendations. We have, however, modified this recommendation, as deemed appropriate, based on your response.

6. Recommendation 1: After further evaluation, we are not insisting that you implement global FAIL Mode at this time. We recognize that there are 28 system

components operating in FACILITY FAIL mode, including the CICS region supporting HUDCAPS. However, we disagree that this has been the case since its inception. From October 1994 to June 1995, the CICS region for HUDCAPS was operating in FACILITY WARN mode. During this time, not only was the HUDCAPS CICS-facility in FACILITY WARN mode but also the system-wide security was in GLOBAL WARN mode. Under FACILITY and GLOBAL WARN mode, CA-Top Secret could not protect HUDCAPS from unauthorized access. Even now, the security has been set to the WARN mode for two facilities, APPC and UNICENTER. HUDCAPS data and program files are exposed to unauthorized access from facilities operating in WARN mode.

7. Recommendation 2: We changed this recommendation to emphasize that only systems with appropriate security be placed into production. HUDCAPS was placed into production without adequate security for a period of six months. Control must be in place to ensure that this practice is not allowed.

8. Recommendation 3: We need documented evidence that the Top Secret vendor does not provide the necessary interfaces with the products needed for data base management and reporting. We also have not received the mentioned cost-benefit analysis for replacing DB2 internal security with Top Secret/DB2 security. Consequently, we cannot agree to a management decision for this recommendation.

9. Recommendation 4: IT may have already implemented this recommendation. We have requested but have not yet received evidence that the individual userid with access to INSTALL SYSADM authority has been removed.

The discussion about ‘audit flags’ attached to DB2 system administrators is not relevant to this recommendation. This recommendation only pertains to controlling the use of the INSTALL SYSADM privilege under DB2 and not to DB2 system administrators.

10. Recommendation 5: Our concern over an audit trace pertains to the granting of any high-level privileges. DB2 provides an audit trail that logs the specific accesses that are granted or revoked. This option is enabled by setting the AUDITST parameter to YES during initialization of the subsystem. DSNZPARMS printout that IT provided shows that HUDCAP’s setting of AUDITST=NO disables security authorization checking in DB2 and disables the tracking of GRANTs. To minimize any impact on system performance, we recommend that IT fine-tune the audit process further by choosing the audit option, which is related to security events only.

11. Recommendation 6: The number of systems programmers (over 20) with update capability should be significantly reduced. Only a minimal number of system programmers, based on job functions, need update capability in a production environment.

12. Recommendation 7: During our field work in the early part of 1996, the HUDCAPS contractor staff indicated that there were no formal documented procedures in place.

13. Recommendation 8: We do not dispute that DB2-secondary IDs are found on the system. However, we question the basis on which users were granted the secondary

IDs. We requested but did not receive an organization chart of employees and their job functions. Without an official organization chart, we could not relate the correspondence between job function access requirements and the DB2-secondary ID groupings.

14. Recommendation 9: We agree with IT's plan to place legacy systems under RDO as they are converted to CICS 4.1 and to configure new systems to use RDO at initiation. The referred recommendation is not in the final report.

15. Recommendation 10: We note that IT has reduced the number of userids granted the NODSNCHK privilege, which bypasses dataset security without an audit log. However, four system userids: CA7DEFT, NOMAD, DFHSM and M1NOPRD1 still retain this privilege. Additionally, there are still some 50 system started tasks with the 'alternate' security bypass privilege (DATASET=*., ACCESS=ALL). Many of these started tasks have been in operation in this manner since 1994, with update-access to all files on the system. IT must remove all bypass privileges for started tasks by December 31, 1997 as planned. Any delay in the implementation would prolong the exposure of HUDCAPS to unauthorized access.

16. Recommendation 11: The system userid assigned to the CICS production region for HUDCAPS has the bypass dataset access checking privilege. Further, users of this system userid can submit batch jobs. This means the 30 plus users permitted this access can update any program or data file on the production system. IT did not specify the reasons why so many users need this access.

17. Recommendation 12: The response did not specify the controls for granting access to powerful CICS transactions. Granting over 70 userids the CEMT transaction privilege exposes HUDCAPS to unnecessary risks of system failures and/or data errors. Only individuals assigned to administer CICS Region for HUDCAPS should be granted the use of this function.

We disagree with IT's statement that CEMT is removed only at the request of the system owner. The HUDCAPS system owner cannot be expected to understand the functions of CICS transactions. IT has the responsibility to administer system software such as CICS.

18. Recommendations 13 and 14: We fully understand the function of CEDA and CEDF transactions and we agree 'that these transactions provide useful tools when debugging and testing ...during development'. However, IT has not followed its own 'procedure to remove the transactions once the application has moved to production.' These transactions have been available to over 70 userids on HUDCAPS production CICS-region.