Payoff

Some RACF security reviews seem to take forever and produce few important results, while others completed in a few weeks result in significant findings. Such extreme outcomes usually reflect differences in methodologies, tools, and skill sets. This article, the first in a two-part series, discusses an approach to reviewing an MVS/RACF system in an effective and efficient way. Part one of this article presents an introduction to MVS/RACF controls and a methodology for a commonsense security review.

Problems Addressed

The problems addressed in this article are the selection of an appropriate approach for the security review and the selection and use of software audit tools. The second part in this series discusses the skills needed to perform an effective security review and specific considerations for the review of three areas: security management, system integrity, and application security.

MVS/RACF Security Concepts

MVS, IBM's operating system for large mainframe computers, offers system integrity, an important property for implementing security controls, but does not have any practical built-in access control functions. These functions are provided by add-on products such as IBM's Resource Access Control Facility (RACF). Security and integrity in this environment is implemented through a combination of MVS and RACF controls. The following sections provide a high level overview of these control areas.

Access Control through RACF

RACF controls provide three basic functions:

- Identification and authentication of users and system access control to ensure that only authorized users are admitted to the computing system and to establish personal accountability for all resource accesses and uses.

- Access authorization checking (i.e., the application of predefined rules as users access or use resources such as data sets and programs).

- Logging and reporting of security-related events (i.e., the creation and processing of audit trails).

In addition, RACF provides user interfaces for security administration and audit. Many RACF functions are primarily used to protect applications. In addition, system-wide RACF functions support system integrity and complement MVS controls. Application-related RACF functions are used to control access to application files, programs, and transactions. These functions are used to implement policies such as least necessary privilege and segregation of duties. System-related RACF functions support policies involving enforced user authentication, default resource protection, and password management.

RACF provides baseline controls, a set of controls considered necessary in most commercial environments. Other RACF controls are considered to be advanced; they are
only used selectively, depending on risk level. For example, implementation of information classification through RACF security labels is an advanced control in most commercial settings.

**System Integrity Controls**

System integrity controls are mechanisms in the operating system and its subsystems designed to separate normal, nonprivileged users and processes from privileged system functions, and prevent them from obtaining system privileges. It is important to restrict privileged system functions because they include capabilities to alter, disable, or circumvent Resource Access Control Facility access controls.

IBM has issued an MVS Integrity Statement which guarantees the confinement of unprivileged users under the condition that the installation implements integrity controls properly. MVS integrity controls are considered part of baseline controls.

**Security versus Integrity Controls.**

There is an important distinction between software security controls and System integrity controls. Security controls are designed so that information owners and administrators can implement access rules for their application resources. Integrity controls are related to MVS operating system and global RACF security options; they are implemented by system support and RACF administration functions to ensure System integrity.

**Control Dependencies**

It is important to understand that security and integrity controls are rarely independent. They usually depend on other controls, often in a hierarchical order, and sometimes in an interdependent way. An effective approach to auditing MVS/RACF should recognize the hierarchical dependency of major control categories, as depicted in Exhibit 1.

**Control Layers**

Software security controls, as a whole, depend on a foundation of management directives, comprised of a security policy, standards, and procedures. Application-level security controls depend on system integrity controls. A security review of application controls alone, without regard to System integrity, might conclude that controls are adequate when, in reality, the system (and therefore all applications) may be open to attack at the system level. In fact, the author's audits of a large number of MVS audits have shown that the majority had severe deficiencies in System integrity controls.

**Application-Level Controls**

Application-level security controls can be described in terms of a matrix of system users (i.e., subjects) and system resources (i.e., objects) containing access rules (see Exhibit 2). Application controls can be implemented within business applications (e.g., general ledger, accounts payable) or application environments (e.g., Internet Multicasting Service, Canadian Independent Computing Services Association) in the form of internal security tables, or these controls can be implemented externally in Resource Access Control Facility. Their implementation in RACF has the advantage of maintaining controls in one secure place.
**Access Matrix**

In RACF, subjects and objects can be listed individually or, to improve administration efficiency, organized in groups. RACF provides the capability to connect users to user groups and to protect groups of resources through generic or grouping profiles. For example, all system data sets with a high-level name qualifier of SYS1 can be RACF protected through a single generic profile named SYS1.

Access rights can be defined at different levels: NONE, EXEC, READ, UPDATE, and ALTER. These access levels are hierarchical, i.e., READ includes the EXEC capability, UPDATE includes READ and EXEC, and so forth. Public access (i.e., access for users or groups not explicitly listed) is defined as universal access (UACC) or ID(*).

Definitions of users, resources, and access rights are entered and maintained through the RACF administration interface, and they are stored in profiles in the RACF database. These definitions are used when RACF is called to determine whether a user's request should be granted or not.

**System Integrity Controls**

Even the best set of application controls may not provide reasonable security if unprivileged system users can bypass or alter access rules or if privileged users are not reasonably controlled. Examples of such potential exposures include:

- Unprotected Authorized Program Facility-libraries or improper Switched Virtual Circuit that allow unprivileged users to execute programs that can defeat (bypass or alter) RACF access controls.

- Uncontrolled RACF privileges that give too many users (and processes) unlimited access to data sets or security administration privileges.

System integrity controls address these threats. To understand System integrity controls, it is necessary to first understand the environment in which programs execute in MVS.

As shown in Exhibit 3, the system space contains the MVS operating system and its subsystems, the RACF security system, and any component authorized by the installation to execute with system privileges. Typical system privileges are supervisor state, a property that allows the execution of privileged instructions, and system key, the use of a storage protect key reserved for system use. APF (APF), the third system property depicted, is the installation-controlled designation of a program that executes with system privilege. All system programs execute at the same level of privilege and can therefore affect each other.

**MVS Integrity**

The user space contains all other programs that do not have system privileges. These programs execute in problem state under a user key and are not loaded with the Authorized Program Facility privilege. MVS integrity controls prevent them from interfering with system code and, through address space segregation, with other applications. These user programs are covered by IBM's MVS integrity statement.

MVS integrity controls are represented by the dividing line in Exhibit 3. When proper controls are in place, application programs will not be able to cross this line and obtain system privileges (which include capabilities to alter or circumvent access controls).

An overall methodology to review system security and integrity controls evaluates:
The options that must be selected in customizing the system.

- The integrity of system catalogs, files, and tables.
- Privileges that must be assigned in a restrictive way.
- Critical functions that exist in an MVS environment and must be carefully restricted to trusted users.
- System extensions and modifications that have a potential for undermining integrity.

These controls are described in more detail later in this article.

**Policy, Procedures, and Standards**

A security policy is needed to set and communicate the general rules and directions. Procedures and guidelines provide a detailed interpretation of the policy for different roles, such as security administrators, resource owners, end users, and auditors. Security standards translate the policy into technical standards, options, and settings in the operating system and the security software.

Without adequate security management, controls are rarely built according to an organization's needs and are usually not kept at the required level. Security and integrity controls tend to deteriorate over time unless they are closely monitored.

It is important to understand that System integrity controls are not protected against deterioration because they are built into the system; on the contrary, they are very much exposed to human actions and errors. Reasons for the observed deterioration of controls include:

- **System maintenance activities.** Critical MVS controls are often affected by software maintenance, release upgrades, and migrations. The activities of separate support groups (e.g., MVS, IMS, CICS, Virtual Telecommunications Access Method) are often not adequately synchronized.

- **Security administration activities.** Temporary access to the system or to resources is often not removed on a timely basis, and Resource Access Control Facility definition and rules may get out of touch with reality (e.g., obsolete user IDs, no rule verification).

- **New products or programs.** New system or application subsystems may introduce security and integrity exposures if they are not properly screened.

**Baseline Controls**

Theoretically, security controls fall into two general categories: baseline controls and advanced controls. Baseline controls must be in place to establish a working set of controls; if one is defective or missing, the whole set is defective. MVS integrity controls are a perfect example of baseline controls; if one control area is exposed, the whole system has an integrity exposure. (An analogy in terms of physical security is that a fence must be complete and all doors have locks to establish a basic level of protection.) In practice, all MVS controls and most Resource Access Control Facility controls should be regarded as baseline controls.

Advanced controls are controls that may be considered for additional security beyond the baseline, usually based on specific risks. (In terms of physical security, advanced...
controls might involve increasing the height of a fence or installing extra-strength locks on doors.) Mandatory access control that can be implemented through RACF security labels, for example, is usually regarded as an advanced control in a commercial environment. “Erase-on-scratch” is mostly viewed as an advanced RACF control in the private sector. The use of cryptography (beyond password encryption) is also considered an advanced control.

**The MVS/Racf Security Review**

This section discusses alternatives and recommendations for the basic security controls review. It includes a discussion of appropriate audit tools and techniques.

**Approach To Reviewing Security Controls**

When MVS/RACF controls are reviewed, the primary decision to use a controls-oriented approach (as opposed to substantive testing) has already been made. Within this approach, additional choices have to be made between a procedural and technical focus, and a comprehensive versus a priority orientation. Authorized Program Facility library controls are used in this section to illustrate these points.

**Procedural versus Technical Focus.**

Procedural controls consist of human actions required to establish or maintain computer controls. Technical controls are computer controls, mostly in software, that work automatically once they are implemented. A balanced audit must address both procedural and technical controls in the following order of importance:

- **Technical evaluation to assess the quality of computer controls.** Quality of controls includes completeness and design; that is, the proper choice of technical functions and features to implement the security policy. This is the most important assessment because it addresses the controls that are active in the computing environment. If they are inadequate they usually cannot be compensated for by procedural controls and therefore need to be fixed immediately. (As an example, the proper protection of all APF libraries is a vital technical control; exposures require immediate corrective action and cannot be limited or compensated for by procedural controls.)

- **Procedural evaluation to assess how computer controls are monitored to ensure that existing controls do not deteriorate over time.** Computer security tends to weaken if not monitored adequately. (As an example, regular monitoring for APF libraries and their protection is an important procedural control.)

- **Procedural evaluation to assess compliance with administrative procedures.** Proper signoff on all security-related documents and similar procedures are part of a well-controlled system, though technical controls are more important than strict procedural formalism. (For example, the assessment of proper paper trails for APF libraries is important, but not as important as the preceding two types of controls.)

**Exhibit 4** shows the importance of a technical controls evaluation based on echo statistics gathered in MVS audits and security review. These statistics use a simple rating scheme to categorize audit findings:

- **Exposure.** Serious control weakness, immediate action required.
Concern. Control weakness, corrective action required.

Housekeeping. Potential weakness, cleanup recommended.

Okay. No weakness identified.

**ECHO Statistics Authorization and Privileges (71 Systems in 41 Installations)**

The exhibit shows the statistics for the category Authorization and Privileges, one of the control categories discussed later in this article. Exposures were found in 59% (42 out of 71) of the reviewed systems.

**Comprehensive versus Priority Orientation.**

Security reviews are performed under time and resource constraints; therefore, controls should always be assessed in order of priority. The dependencies discussed earlier in this article are also reasons for using a prioritized approach.

Not all controls are of equal importance within a controls layer for reasons of criticality and maintainability. For example, statistics show that certain controls for MVS/RACF system integrity are often more poorly maintained than others. A prudent audit approach will use such experience to establish priorities. Exhibit 5 shows the ECHO rating for different controls in the category Authorization and Privileges.

**ECHO Statistics Authorization and Privileges (71 Systems in 41 Installations)**

Some audits tend to take a comprehensive approach, which can actually result in a waste of resources and a false sense of comfort. For example, the complete review of the contents of all APF libraries in an MVS system may appear to be necessary. This is true only in principle and under the following conditions:

- The controls governing the APF environment are reviewed first and are found to be adequate.

- The characteristics of vendor programs are known to the reviewer through a secure channel so that code corruption can be identified with certainty.

- Reasonable tools are available to assist in this process.

Experience shows that the first two conditions are usually not met. Controls over APF are usually not sufficient to rule out the possibility that other libraries are added to the environment without a proper audit trail. The characteristics of vendor code are not known to the reviewer. Freezing libraries at one point in time and identifying changes at a different time are not sufficient; illegal code may have existed already at the first control point, and software maintenance activities may make it difficult to distinguish between authorized updates and unauthorized changes.

In summary, the recommended approach with respect to evaluation of technical and procedural controls is:
· Evaluate the Technical Quality of MVS/RACF controls. Extraction and evaluation tools should be used to test their actual technical status and quality.

· Evaluate their robustness over time. Monitoring and verification procedures should be in place to identify weaknesses and prompt corrective action.

· Evaluate their procedural and management aspects. Procedures should be in line with policy and there should be proper authorization trails.

The recommendation regarding use of a prioritized or comprehensive review approach is to:

· Prioritize audits according to the hierarchy of controls. System integrity should be considered a necessary prerequisite for application controls.

· Prioritize tests according to criticality of controls. Primary controls (e.g., APF libraries) should be addressed before secondary controls.

· Prioritize tests according to maintainability of controls. It should be recognized that APF library controls tend to be more volatile than controls over user Switched Virtual Circuit.

Tools and Techniques

Tools and techniques for the security controls review consist of:

· Checklists and audit programs.

· Extraction software.

· Evaluation software.

This section discusses each of these tools.

Checklists and Audit Programs

Checklists and audit programs are tools designed to guide the reviewer through a section of the security review. They are usually designed to ensure consistent coverage of a subject area and often provide a framework for documenting observations, test procedures, and audit findings.

Checklists and audit programs can be valuable tools if they are designed and used properly. They can be ineffective and even dangerous if:

· They have design deficiencies. Such deficiencies can result if a checklist is a random collection of questions rather than a well designed, comprehensive set of questions covering the subject matter. In addition, an outdated checklist (e.g., reflecting an old release of a software product) may not be effective.

· They are used by the wrong person. A checklist is not a substitute for user skill and knowledge. It is only a good tool in the hands of an educated user who understands but does not simply memorize.
They are used inappropriately. Even well-designed and up-to-date audit programs may be more comprehensive than the individual situation might justify or require. They should be used selectively, following priorities discussed earlier in this article.

In summary, checklists and audit programs can be very useful, particularly if they are computer-based. They can be used in conjunction with software tools like the ones discussed below.

**Extraction Software**

Extracting MVS/RACF controls manually is very time consuming and requires system-programmer-level knowledge and skills. Effectiveness and efficiency considerations demand that extraction software be used in an MVS/RACF security review. These tools enable the reviewer to document and understand MVS information in a convenient way, without the need for system programmer knowledge. The Resource Access Control Facility tools provide audit functions far beyond the capabilities of basic RACF commands and utility programs. A variety of products exist; the following are products with which the author has significant practical experience.

**DSMON.**

This utility program, which is supplied by IBM as part of RACF, reports on selected RACF and MVS controls. It should be used in combination with the RACF command SETROPTS LIST. DSMON has been available for 10 years; it was the first tool in this category. Unfortunately, it has not been enhanced by IBM as new RACF releases introduced new functions and features.

**CA-Examine.**

CA-Examine, a product of Computer Associates, is a standard tool for MVS audits. It can be used to extract a great deal of MVS control information, and it has other useful operational functions. Although it is a useful product, it does have some functional limitations. It does not report on access controls provided by RACF, CA-ACF2, or CA-Top Secret. It also does not run authorized; therefore, it cannot access protected MVS control blocks (which contain the most reliable information).

**Vanguard RACF Administrator.**

Vanguard Integrity Professionals' Vanguard RACF Administrator offers a great variety of RACF administration and audit functions. The audit reports present RACF definitions and rules in a user-friendly format and provide the reviewer with a good set of standard reports. Selected MVS controls are also addressed.

**Consul/RACF.**

Consul Risk Management's Consul/RACF offers a comparable set of standard RACF administration functions and audit analysis. In addition, it provides two major features:

- Collection of comprehensive resource information (e.g., from Volume Table of Contents, catalogs, Partitioned Data Set directories) and reporting functions for rules and resources.
A report generator facility that enables the reviewer to develop custom reports.

Evaluation Software

Some software tools are also capable of evaluating controls (within certain limitations). Vanguard RACF Administrator offers an evaluation of system protection, and Consul/RACF offers similar services plus a variety of verification functions that correlate rules and corresponding resources, and report discrepancies. It can be expected that more tools of this sort will be developed to support evaluation of controls.

Recommended Course of Action

An effective security review of an MVS/RACF environment requires that the reviewer have a firm understanding of how operating system controls interact with and affect access controls. Security and integrity controls are rarely independent. Therefore, an effective approach to auditing MVS/RACF should recognize the hierarchical dependency of application-level controls, system integrity controls, and associated policies, procedures, and standards.

Because security reviews are performed under constraints of time and resources, it is recommended that controls always be assessed in order of priority. Various software-based tools are now available to assist the reviewer in assessing controls. It is important that the available tools be carefully selected to meet the objectives of the review.

Author Biographies

Kurt Meiser

Kurt H. Meiser is director of ITSS International, Inc., a consulting firm based in Poughkeepsie NY that specializes in computer security. Previously, Meiser was a manager at Coopers & Lybrand in New York for six years, with responsibility for design, development, and security of information technology security services. Before that he was systems engineer for IBM Corp. for 22 years, with emphasis of MVS and RACF integrity and security.
<table>
<thead>
<tr>
<th>Subjects</th>
<th>USER1</th>
<th>USER2</th>
<th>GROUPA</th>
<th>PUBLIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects</td>
<td>PAY: **</td>
<td>UPDATE</td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TEST: **</td>
<td>ALTER</td>
<td>READ</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PROGRX</td>
<td>READ</td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRAN1</td>
<td>READ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TGRPA</td>
<td>READ</td>
<td>NONE</td>
<td></td>
</tr>
</tbody>
</table>
Note:
APF = APF libraries, TSO = TSO authorization tables, PPT = Program Properties Table, USR = RACF user privileges