CA-7®

Security Guide

3.3

COMPUTER ASSOCIATES®
Software superior by design.

First Edition, September 2000

Contents

Chapter 1. Introduction

This manual describes the steps necessary to implement CA-7 security.

Note: Because this document contains sensitive information, it is recommended that distribution of this guide be limited to those responsible for implementation and maintenance of CA-7 security.

1.1 Summary of Revisions

This topic explains changes to both CA-7 and to the documentation.

1.1.1 Product Changes

CA-7 Version 3.3 contains the following major enhancements:

- Parallel Sysplex Exploitation

  CA-7 can optionally maintain a memory structure in the Coupling Facility in which participating ICOMs record tracking data. One or more Host ICOM(s) read from the memory structure and write to the Communication data set. This can significantly reduce I/O contention and increase feedback throughput.

- UNIX System Services Interface

  The OS/390 UNIX System Services (USS) CA-7 interface allows communication with CA-7 from the USS environment. The interface can be called directly from the UNIX shell or from the IBM USS batch interface (BPXBATCH).

- CA-7 CCI Interface

  The CA-7 CCI interface allows two-way communication with CA-7 from other address spaces and environments. The interface can be engaged in a batch mode, in a REXX address environment or it can be called directly from a user program. It accepts single or stacked commands as input and returns the CA-7 output from the commands as if they had been executed in batch mode.

- Critical Path Monitoring

  Through integration with CA-OPS/MVS II, Unicenter TNG and Unicenter TNG MVS Event Manager Option (MEMO), CA-7 can support the definition and monitoring of critical job flows within the CA-7 workload. CA-OPS/MVS II provides management and administration of critical path displays.

- Mixed Case Support in CA-7 Editor

  Character translation controls can be set in the CA-7 Editor. New Editor subcommands 'UPPER' and 'MIXED' determine whether editor data is translated to uppercase or left "as is."

  These subcommands are enabled with a new initialization file option. If this option is not coded, then all edit data is translated to uppercase.

- Job Completion Tracking Precision

  CA-7 records job completion times in hundredths of seconds. This allows job completions to be discriminated with a high degree of precision, thus reducing the likelihood of requirement posting ambiguities where jobs complete within the same minute.

- Display Duplicate Days for RESOLVe

  CA-7 can optionally display the duplicate RESOLV day(s) in new message SRC1-137. This occurs when a job is scheduled to execute the same day under two or more different Schedule IDs. With this information one can more quickly and efficiently determine the source of the scheduling conflict.

- VRM Device Control

  Virtual Resource Management (VRM) Device Control provides an alternative to Workload Balancing control of job submission based on tape drive availability. VRM resource count resources representing the number and type of storage devices used by the job are defined dynamically during CA-7 LOAD processing.

  Workload Balancing only permits two types of tape drives. With VRM Device Control, the number and structure of device groups is determined by the user.

- CA-7 Command Retrieval

  Command line input for CA-7 VTAM terminals is recorded in storage and may be retrieved with the /FETCH command. When the /PFnn command is used to associate /FETCH with a PF key, the CA-7 user can conveniently retrieve the last five CA-7 commands entered at an online terminal.

- CA-7 Base Calendar Security

  CA-7 security can allow clients to define CA-7 base calendar names to an external security product and secure user access to individual base calendars.

- REXX Address Environment

  Using the new CA-7 CCI interface, CA-7 allows REXX programs to pass commands to CA-7 and take action based on the output from those commands.

- Job 'Purge' Function

  The DB.1 (Job) panel provides a new function, PURGE, which deletes all CA-7 database records related to a job. In addition to the standard delete processes, the PURGE function deletes incoming trigger definitions, requirement successor definitions, and the CA-11 CMT member for the job.

- Suppress LATE Designation

  Through an Initialization File option, the PROMPTS field on the DB.1 (Job) panel can be used to indicate certain jobs should never be marked as LATE on status displays. This means operations and production control staff will not be distracted when test or non-critical jobs do not complete on time.

- CSA Chains Above the 16M Line

  CA-7 CSA SMF and Trailer chains now reside in extended CSA (above-the-line), thereby reducing utilization of this critical resource.

- Automated Recovery Facility (ARF) Enhancements

  CA-7 can optionally add a LOGON parameter to the ARF TSO SEND command to cause messages to be retained until the user logs on to TSO. Also, support for ARF has been added to the Database Transportability facility.

- Prior Run Queue Expansion

  The maximum size of the Prior Run Queue is now approximately twice as large as in prior releases.

- CA-7 JCLCheck Common Component

  The CA-JCLCheck Common Component is provided in place of the CA-7 JCL syntax checker.

- Documentation Files on Tape

  The current CA-7 documentation files are provided in IBM Book Manager and PDF format on the product tape.

- Other Enhancements:

  – SMF Purge records may optionally be sent to a test copy of CA-7. This allows detection of pre-execution JCL Errors by the test copy.

  – The Scratch and Disk Queue Table queues can be formatted during a CA-7 ERST start which facilitates use of VIO to improve performance.

  – The LJOB command provides a new option, LIST=RQEXCP, that lists only those requirements with a SKIP or ONLY indication.

  – The reverse forecast commands, FRJOB and FRQJOB, have a new option, LIST=HDRS. This will limit the display to only the target job and all 'header' jobs.

  – Database Transportability now supports a new keyword, NODSNS, for SASSDT30 which prevents the generation of data set definitions.

  – The LQ family of commands (LREQ, LRDY, LACT, and so forth) now support a Schedule ID filter, SCHID=.

  – The LRLOG command has a new sequence option, SEQ=REV, which causes entries to be displayed in reverse date/time sequence (most recent first).

  – The OPTIONS initialization file statement has a new keyword DPROCCOM= to enable comment statements in CA-Driver procedures.

  – The OPTIONS initialization file statement has a new keyword EXTSCHID= to set a default schedule ID for externally tracked jobs that are not assigned a non-zero schedule ID from the SASSEXTT table.

  – The CA-7 CAIRIM initialization module now accepts a new reinitialization parameter (REINIT=UTABS) to reload only user defined table modules.

  – The /DISPLAY command has a new STATUS option (/DISPLAY,ST=CA7) to describe the current copy of CA-7 (VTAM application ID and so forth).

1.1.2 Documentation Changes

The documentation for CA-7 Version 3.3 differs from previous releases as follows:

- The documentation set has been engineered to take advantage of the latest technology for online viewing, keyword searching, book marking, and printing. The set consists of a hard copy *CA-7 Getting Started* guide and Version 3.3 of CA-7 for OS/390 documentation in both IBM BookManager and Adobe Acrobat Reader format on the tape.
- Unicenter TNG Framework for OS/390 is composed of the services formerly known as CA90s and Unicenter TNG Framework.
- Reading Syntax Diagrams in the *CA-7 Commands Guide* explains how to read the command syntax used in all guides.

Technical changes are identified by a revision bar (|) in the left margin. Revision bars are not used for editorial changes and new manuals.

1.2 Security

CA-7 security provides a control structure which enables each installation to protect data and resources being accessed through CA-7. Two options are available when implementing CA-7 security:

- external security interface
- CA-7 internal security

Use of external security to control access to CA-7 resources allows maintenance of security to be centralized. Existing USERID definitions and data set access rules may be extended for use in the CA-7 environment if the external security interface is used.

CA-7 internal or "native" security can be used to control data and resources which are unique to the CA-7 processing environment.

Chapter 2. Implementation Considerations

This chapter identifies implementation considerations including identifying your security requirements and CA-7 security structure and cross-platform scheduling security considerations.

2.1 System Requirements

The CA-7 External Security Interface is available with CA-7 Version 3.0 and above.

The CA-7 External Security Interface requires Unicenter TNG Framework for OS/390 CAIRIM and CAISSF at genlevel 9212 or above. (Unicenter TNG Framework for OS/390 is composed of the services formerly known as CA90s and Unicenter TNG Framework.)

For CA-ACF2 installations, the CA-7 External Security Interface requires CA-ACF2 Version 5.2, genlevel 9007 or above.

For CA-Top Secret installations, the CA-7 External Security Interface requires CA-Top Secret Version 4.2, genlevel 9006 or above. Genlevel 9006 requires PTF CO54645.

For RACF installations, the CA-7 External Security Interface requires RACF Version 1.7 or above.

2.2 Identifying Your Security Requirements

The user can define the security structure and level of authority for each individual accessing CA-7. A careful evaluation should be made of your security requirements prior to implemention of CA-7 security.

When a variable number of people are involved in handling the work flow, some method must be used to control who performs specific tasks and an authorization level must be defined to protect your data and resources. Facilities provided by CA-7 for accessing and maintaining the database, as well as monitoring and controlling the production process, require performance of specialized tasks by different people with varying levels of responsibility.

To ensure integrity of the CA-7 system, provisions have been made to allow each installation to define a control structure or hierarchy required within their environment. Levels of responsibility are then assigned to specific individuals depending on that individuals activities within the hierarchy.

When defining the control structure, the following considerations should be made:

1. Which security options will be used?
2. Who can access and use CA-7?
3. Do individual users need access to only certain commands or applications under CA-7?
4. On formatted screens, should users be restricted from certain functions (Add, Update, Delete)?
5. Will USERIDs exist in the JCL for a job or will the USERIDs be inserted by CA-7 or a user exit?
6. Can users issue commands or submit jobs which contain a USERID other than their own?

2.3 CA-7 Security Structure Using External Security

The CA-7 External Security Interface provides key functional control points which are used to determine access authority for users of CA-7. The control points are as follows:

1. Logon
2. Data Set
3. Command Authority
4. Panel Access
5. Job Submission
6. External Communicators
7. USERID Protection
8. Multi-CPU Security Environments
9. CA-7 Submit Data Sets

When access to a given resource is attempted, a security check is performed to determine the authority of the user initiating the request. The user's authority to access the data or resource is determined by the security definitions in place at the time of the access. Refer to the following topics for a description of the control points.

2.3.1 Logon

Logon Security provides a means to control access to CA-7 and its facilities. Each user requesting access to CA-7 requires validation by the external security system. Assuming that the logon information is valid, the user is signed on to CA-7. One additional step in the logon validation process can be implemented. By defining CA-7 to the external security package as resource, an additional check is made, after the user is validated, to determine the user's authorization to access the CA-7 resource. There are some differences between security packages when implementing this secondary check for the LOGON process. Refer to the appropriate chapter in this guide for specific details that relate to the security package you have installed at your site.

Note: To use any of the CA-7 External Security Interface control points through external security, Logon security must be implemented. Logon security establishes the security environment for each user of CA-7 and allows communication with the external security package.

The use of the MVS console from CA-7 (using the modify command) requires a logon userid, but no password is required.

2.3.2 Data Set Security

Data set security is used to control access to data sets by users signed on to CA-7. Each attempt to access data through CA-7 will be validated by security. Access authorization to data sets through CA-7 should not differ from the access you have defined for users outside of the CA-7 environment. Access is done under the CA-7 USERID.

2.3.3 Command Authority

Command security provides a means to control access to the wide range of command options available under CA-7. Because CA-7 offers some very powerful commands, it is very important that the command authority for each user of CA-7 be restricted to individual areas of responsibility.

2.3.4 Panel Access

Panel security is provided to control access to the various application menus and panels throughout CA-7. Each panel within CA-7 has a unique panel-ID which can be used to restrict access based on each user's area of responsibility. Additionally, any functions which appear on the panels can be restricted by specifying access levels, such as READ and WRITE, for each panel.

2.3.5 Job Submission

External security packages generally require a USERID and password to be associated with every batch job which executes on the system. Based on special keywords, specified in the CA-7 initialization file, you can set up a hierarchy of candidate USERID sources from which CA-7 selects IDs for USERID insertion prior to submission.

2.3.6 External Communicators

The External Communicators (SASSBSTR, SASSTRLR, U7SVC, and SASSBCLP) provide a means for users outside the CA-7 address space to send terminal transactions or post data set creations to CA-7. Refer to the chapters of this guide that specifically target implementation considerations for your security environment (that is, CA-ACF2, CA-Top Secret, or IBM RACF) for more information on the secure use of the External Communicators.

2.3.7 USERID Protection

In the standard security environment, each user is assigned a USERID which restricts the user's access to resources related to his or her areas of responsibilities. To prevent users from using a USERID other than their own, submit checking can be implemented to restrict access to USERIDs. Submit checking is performed for the following conditions:

1. Requests for jobs, such as DEMANDs, LOADs, or RUNs when the job's JCL contains a USERID.
2. Attempts to add USERIDs to JCL through the CA-7 text editor or queue JCL editor.
3. Attempts to add, update, or delete the OWNER field associated with a job on the CA-7 DB.1 screen.
4. Executions of the SASSTRLR or SASSBSTR facilities when a USERID is supplied on a /LOGON card. If the USERID supplied is different than the USERID identified in the external environment, a submit check is performed to validate the user's authority to submit for the supplied USERID.

For users to use a USERID other than their own, specific authorization must be granted through the external security package.

2.3.8 UID Assignment

The UID assignment for users of CA-7 can now be controlled through external security using UID Resources. A UID Resource Table (default - SASSRTBL) allows sites to define a resource name to UID value relationship which can be validated through external security during logons to CA-7. This eliminates the need to maintain the CA-7 internal security module with all USERIDs and provides the UID level security assignment to be controlled by external security. See the section on the appropriate security package for specific details on implementation of UID Resources.

2.3.9 Multi-CPU Security Environments

CA-7 identifies the external security processing environment during initialization on the host CPU. The format for JCL USERID insertion, during job submission, is dependent on the security package present at startup. CA-7 does not support multiple security environments for remote job submission due to the JCL USERID format restrictions imposed by the individual security packages.

2.3.10 CA-7 Submit Data Sets

The CA-7 Submit data sets are used in nonshared spool, shared DASD, multi-CPU MVS systems. The JCL is written to the Submit data set by CA-7 and then read by CA-7 ICOM for submission on the appropriate system. With USERID insertion during job submission, the JCL USERID format is determined based on the external security environment CA-7 identified during initialization. Due to JCL USERID format restrictions related to each external security package, CA-7 does not support multiple security environments for USERID insertion.

Note: For SAF compatible systems, such as RACF, CA-7 ICOM does not support the use of Submit data sets and USERID propagation.

2.4 CA-7 Security Structure Using Internal Security

Five distinct levels of security may be defined using CA-7 internal security:

- Terminal/Operator
- Operator/CA-7 Application
- CA-7 Application/Command
- Command/Function
- UID/External Data Set

Refer to Chapter 7, “CA-7 Internal Security” for additional information.

2.5 Security Considerations for CA-7 ARF

ARF for CA-7 allows automation of customized recovery procedures for production jobs. The recovery procedures for an ARF condition may include: sending a message to a specified TSO user or to the MVS console, executing CA-7 commands or scheduling and tracking special recovery (ARFJ) jobs. Refer to the chapter devoted to ARF in the *CA-7 Database Maintenance Guide* for more information.

In the definition of an ARF condition, you may code up to seven recovery action statements. When an ARF condition is detected, CA-7 scans the ARF definition to determine the recovery actions to process.

Each recovery action statement (except AW statements) may cause one or more CA-7 terminal commands to be executed. Although the format of the command must follow conventions used for batch terminal or SASSTRLR input, neither of these terminal types is used for ARF responses. Instead, ARF requires that one or more terminals be defined in the CA-7 initialization file as DEVICE=TRXDV. These TRX terminals are similar to the trailer terminal (DEVICE=TRLDV), except they are dedicated for use by functions internal to CA-7 such as ARF.

Because ARF recovery actions are processed as CA-7 terminal commands, a valid logon is required for each ARF recovery transaction. The ID for this logon is supplied on the AR.3 panel in the RESPONSE-ID field. This ID must have the authority for all transactions that are executed in response to the ARF condition with which they are associated. If several TRX terminals are coded in the CA-7 initialization file then the ID must be valid for ALL of these terminals because ARF transactions may be scheduled on any of these terminals.

If the RESPONSE-ID does not have the authority to execute the responses in its ARFSET then ARF recovery will not be handled properly.

2.6 Security Considerations for Cross-Platform Scheduling

CA-7 has the capability to send and receive job requests to and from other Computer Associates scheduling systems on a variety of platforms. Cross-platform scheduling with CA-7 is documented in the *CA-7 Interfaces Guide*. This section reviews the security considerations of cross-platform scheduling and points you to relevant documentation in this and other CA-7 guides.

2.6.1 CA-7 as Cross-Platform Client

When CA-7 sends a request to a Computer Associates scheduling system on another platform, it is acting as a cross-platform client. This process is fully documented in the *CA-7 Interfaces Guide*. The primary security considerations relate to the CA-7 cross-platform SUBMIT function.

1. The MVS USERID under which the CA-7 cross-platform tracking function runs must have both READ and UPDATE access to the CA-7 XPS PROFILE partitioned data set. It creates and updates a member for each remote system to which cross-platform requests are sent.
2. All MVS USERIDs under which the CA-7 cross-platform submit function runs must have READ access to the CA-7 XPS PROFILE partitioned data set. It reads member CACCENV for global submit parameters. If the SYSIN data for a particular submit job is in a distinct data set, the MVS USERID under which the submit job runs must have READ access to it.
3. CA-7 cross-platform submit jobs are defined, scheduled, and/or demanded in the same manner as any other CA-7 batch job. The MVS USERID under which the batch submit job runs is assigned in the same manner as your other CA-7 batch jobs.
4. A USERID is always passed to the target system. You can specify the USERID explicitly by the SUBUSER parameter in PROFILE or SYSIN. If no SUBUSER parameter is specified, the MVS USERID under which the batch job is running is extracted and used as a default for SUBUSER.

5. A security call may be made to the external security system to determine if the MVS USERID under which the batch job is running is authorized to submit on behalf of the USERID specified in the SUBUSER parameter. This is the same Submit Check that may be done in CA-7 (refer to the specific chapter for your external security system to define rules for Submit authorization). If the SUBUSER parameter is more than eight characters, the value used for the Submit Check is the first eight characters of SUBUSER.

   The Submit Check security call is made under the following circumstances:

   a. If the MVS USERID under which the batch job is running does not exactly match the SUBUSER USERID,

      - and -

   b. Either the BSUBCHK bit in ICMDSECT is ON or the SUBCHECK=YES parameter is set in the CA-7 XPS PROFILE member CACCENV. The BSUBCHK bit in ICMDSECT controls CA-7 security checking for processes outside of the CA-7 address space (BTI, U7SVC, and so forth). The SUBCHECK= parameter cannot be used to suppress Submit Checking if the BSUBCHK bit is on.

      If the return code from the external security system indicates that the MVS USERID does not have authority to submit on behalf of the SUBUSER USERID, the cross-platform request is NOT sent to the target system and the batch Submit job fails.

6. If the USERID assigned to SUBUSER is the value ROOT, an additional security check is made. The SUBROOT= parameter in the CA-7 XPS PROFILE member CACCENV must be set to YES to authorize use of the ROOT USERID. The USERID 'ROOT' has special meaning on UNIX platforms and should be tightly controlled. If SUBROOT=YES is not specified, the cross-platform request is NOT sent to the target system, and the batch submit job fails.

7. The target system may require that you supply a password with the USERID specified in SUBUSER. Use the SUBPASS parameter to specify a password to the target system. The MVS system makes no check to validate the password. To prevent unintended mismatches of USERIDs and passwords, the SUBPASS parameter is only honored if it comes from the same source as the SUBUSER parameter. That is, both SUBUSER and SUBPASS must be in the SYSIN data, or they must both be in the PROFILE data.

   The password value is encrypted before being sent across the network with the cross-platform request. If you want to use passwords, we recommend you specify the SUBUSER and SUBPASS parameters in a distinct file pointed to by the SYSIN DD statement. You can use the standard facilities of your MVS security system to secure this file for READ and WRITE access.

2.6.2 CA-7 as Cross-Platform Server

When CA-7 receives a cross-platform request from a Computer Associates scheduling system on another platform, it acts as a cross-platform server. This process is fully documented in the *CA-7 Interfaces Guide*. The primary security considerations relate to the assignment of a CA-7 USERID under which the requested CA-7 job will be initiated.

1. Cross-platform requests sent to CA-7 may or may not have an explicit USERID sent with it. If an explicit USERID is sent, it may or may not have an associated password sent with it.

   If a USERID is sent with a request, you can require that a password also be sent based upon the system that sent the request and/or the USERID that is specified on the request. Refer to the subsection Cross-Platform Server Password Requirements in the *CA-7 Interfaces Guide* for documentation on controlling password requirements. If a request does not satisfy the password requirements, it is rejected (submit failure) by the Cross-Platform Router before it is passed to CA-7 itself.

2. If no USERID is sent with a request, it may be assigned a default CA-7 USERID based upon the setting of the XPSSID= keyword on the CA-7 SECURITY statement. Refer to 3.1, “SECURITY Statement” on page 3-2 for further information.

3. Once a cross-platform request is passed to CA-7, processing takes place based upon the explicit or defaulted CA-7 USERID. This USERID is 'logged onto' a CA-7 internal terminal. If a password was passed with an explicit USERID, it is specified on the logon command. This logon is handled the same as other logons in your CA-7 system. That is, it is handled using internal or external security based upon the global parameters you have specified on your CA-7 SECURITY statement. If the logon is successful, the CA-7 job specified in the cross-platform request is initiated using a DEMAND or RUN command and is subject to the security restrictions defined for the USERID under which the command is being issued.

2.7 Summary

The implementation and structure of CA-7 security differs based on your choice of internal or external security. Refer to the appropriate chapter in this guide for a discussion of each of the various security options to determine which options meet your installation's security requirements.

Chapter 3. CA-7 Security Initialization Options

This chapter details the CA-7 SECURITY statement.

3.1 SECURITY Statement

The CA-7 initialization file contains control statements which define the processing configuration of CA-7 during startup. The SECURITY control statement determines the security environment for CA-7 based on user-selected keywords.

3.1.1 Syntax

```
SECURITY

►►──SECURITY,NAME=SASSSECI──┬────────────┬──────────────────────────────────►
                            └─,APPL=CA7──┘

►──┬─────────────────────────┬──┬──────────────────────┬──────────────────►
   │           ┌─,─────┐     │  │            ┌─YES─┐   │
   └─,BYPSEC=(─▼─┬─1─┬─┴─)───┘  └─,DISPLAY=──┴─NO──┴───┘
                 ├─2─┤
                 └─3─┘

►──┬───────────────────────────┬──┬──────────────────────┬────────────────►
   └─,EXTERNAL=──┬─CALENDAR─┬──┘  │            ┌─NO──┐   │
                 ├─COMMAND──┤     └─,HIDEGRP=──┴─YES─┴───┘
                 ├─DATASET──┤
                 ├─EXTERNAL─┤
                 ├─LOGON────┤
                 ├─SUBCHECK─┤
                 └─SUBOWNER─┘

►──┬─────────────────────┬──┬──────────────────────┬──────────────────────►
   │          ┌─NO──┐    │  │            ┌─NO──┐   │
   └─,HIDEPW=─┴─YES─┴────┘  └─,HIDEUPD=──┴─YES─┴───┘

►──┬───────────────────────┬──┬─────────────────────┬─────────────────────►
   │            ┌─NO──┐    │  │           ┌─YES─┐   │
   └─,HIDEUSER=─┴─YES─┴────┘  └─,JCLUID=──┴─NO──┴───┘

►──┬───────────────────────┬──┬──────────────────────┬────────────────────►
   │            ┌─YES─┐    │  │            ┌─YES─┐   │
   └─,LOADSUBC=─┴─NO──┴────┘  └─,LOGOPID=──┴─NO──┴───┘

►──┬────────────────────────┬──┬──────────────────────┬───────────────────►
   │             ┌─NO──┐    │  │            ┌─YES─┐   │
   └─,PROPAGATE=─┴─YES─┴────┘  └─,PSOWNER=──┴─NO──┴───┘

►──┬──────────────────────┬───────────────────────────────────────────────►
   └─,RCLASS=xxxxxxxx─────┘ ┌────────────────────────┐
                            │              ┌─YES─┐   │
                            └─,RESLOGON=───┴─NO──┴───┘

►──┬──────────────────────┬──┬──────────────────────┬─────────────────────►
   │           ┌─YES─┐    │  │            ┌─YES─┐   │
   └─,RLOGUID=─┴─NO──┴────┘  └─,STATSID=──┴─NO──┴───┘

►──┬──────────────────────┬──┬─────────────────────────┬──────────────────►
   │           ┌─NO──┐    │  └─,SUBUID=──┬─OWNER─┬────┘
   └─,SUBNOID=─┴─YES─┴────┘              ├─REQ───┤
                                         ├─QJCL──┤
                                         └─CA7───┘

►──┬────────────────────┬──┬──────────────────────┬───────────────────────►
   └─,UID=xxxxxxxx──────┘  └─,USER=xxxxxxxx───────┘

►──┬─────────────────────────────────┬───────────────────────────────────►◄
   │          ┌─**NONE**─┐           │
   └─,XPSSID=─┴─xxxxxxxx─┴───────────┘
```

Where:

SECURITY
Identifies the CA-7 SECURITY statement which describes the CA-7 security environment.

NAME
Identifies the load module which contains the CA-7 Security definitions built using the SECURITY macro. **This parameter is required for both internal and external security**. If you are implementing full external security control for CA-7, the default security module SASSSECI can be used. This module is supplied in the CA-7 Load library. If you are using CA-7 internal security, refer to Chapter 7, “CA-7 Internal Security” on page 7-1 for a discussion on building and modifying the CA-7 security module to meet your installations security requirements.

APPL
Identifies the CA-7 Security Application ID. The application ID, if specified, is used as an additional check during Logons. A Resource check is performed, using the Security Application ID as the resource name, to validate the user's authority to access CA-7.

BYPSEC
Specifies functions for which UID security is to be bypassed when accessing jobs in the CA-7 database or queues.

> **Warning**
>
> Use of these options is NOT recommended. If selected, serious security exposures may result. The decision to use these options should be made only after careful consideration of the possible consequences of bypassing the CA-7 security interface.

If more than one of these subparameters is used, enclose in parentheses and separate them with commas.

1
Indicates that security access to predecessor jobs will not be validated during job predecessor definition (DB.3.2). Access to the job for which predecessors are defined will still be validated.

2
Indicates that security access to jobs will not be validated during forecast processing.

3
Indicates that security access to requirement successor jobs is not validated during job 'purge' delete processing (DB.1). That is, if a user is deleting job A with the PURGE function and job B has job A listed as a predecessor, job B is updated to remove the predecessor entry for job A without a security check to determine if the current user has update access to job B.

DISPLAY

Determines whether the USERID is displayed when it is entered on the Logon screen.

YES

Indicates that the USERID is displayed. This is the default.

NO

Indicates that the USERID is not displayed.

EXTERNAL

Identifies the security functions (calls) which are to be controlled by external security. Required for external security. Any security functions that are not specified on the external keyword are controlled by CA-7 internal security. This would require the presence of a CA-7 security module built using the CA-7 SECURITY macro. Refer to 7.3, “SECURITY Macro” on page 7-10 for more information.

If more than one of these subparameters is used, enclose in parentheses and separate them with commas.

CALENDAR

Indicates that any attempt to access a CA-7 base calendar is validated through external security. The security resource class used for such validation is CALENDAR.

COMMAND

Indicates all command functions are validated through external security. This includes panel access throughout CA-7.

DATASET

Indicates that any attempt to access a data set while signed on to CA-7 is validated through external security.

LOGON

Indicates that logons to CA-7 are validated through external security. This parameter is the minimum requirement for the EXTERNAL keyword when implementing external security. The security calls to external security during CA-7 LOGONs establishes the security environment for each CA-7 user.

SUBCHECK

Validates the usage of USERIDs under CA-7. Requires that SUBUID be coded.

When a user requests a job through a DEMAND, LOAD, or RUN, CA-7 attempts to determine the USERID with which the job runs. External security is used to validate the requesters authority to submit for that USERID. Submit checking is also performed during both JCL and QJCL edits. If a user attempts to add a USERID to the JCL, the user's authority to submit for that ID is checked. This prevents unauthorized usage of USERIDs.

SUBOWNER

Performs that same function as Submit checking except that it relates only to the OWNER ID associated with a job. If a job has an OWNER ID defined, validation is performed for attempts to add, change, or delete the OWNER ID.

If this option is used, CA-7 will also validate attempts to add, change or update the RESPONSE ID associated with an ARFSET.

HIDEGRP

Optional. Used to cause user security group values coded in JCL statements to be overlaid with @ characters whenever JCL is listed with one of the inquiry commands.

NO

Indicates the values are to be displayed. This is the default.

YES

Causes the following to be hidden in inquiry output:

GROUP keyword value in JOB statements

Refer to keyword HIDEUSER for a list of the affected inquiry commands.

HIDEPW

Optional. Causes user security password values coded in JCL statements to be overlaid with @ characters whenever JCL is listed with one of the inquiry commands.

NO

Indicates the values are to be displayed. This is the default.

YES

Causes the following to be hidden in inquiry output:

```
PASSWORD keyword value in JOB statements
//*PASSWORD statement values
```

Refer to keyword HIDEUSER for a list of the affected inquiry commands.

HIDEUPD

Optional. Used to suppress the last updater on several of the listing commands.

Note: After enabling this, an update will need to be performed on the element. When this element is updated, *SECURE* will be placed in the updater field.

NO

Indicates values are to be displayed. This is the default.

YES

Causes *SECURE* to be placed in the updater field after the next update.

HIDEUSER

Optional. Causes user security ID values coded in JCL statements to be overlaid with @ characters whenever JCL is listed with one of the inquiry commands.

NO

Indicates the values are to be displayed. This is the default.

YES

Value of YES causes the following to be hidden in inquiry output:

```
USER keyword value in JOB statements
//*LOGONID statement values
//*JOBFROM statement values
```

These values are hidden when the following inquiries are used:

LACT,LIST=JCL

LJCK

LJCL

LLIB

LPDS

LPRRN,LIST=JCL

LQ,LIST=JCL

LQUE,LIST=JCL

LRDY,LIST=JCL

LREQ,LIST=JCL

JCLUID

This option can be used to prevent submission of jobs whose JCL contains a USERID. This parameter is only applicable if a SUBUID hierarchy is also specified. This is an optional parameter and defaults to YES.

YES

Indicates that CA-7 will submit the job after validating that CA-7 has authority to submit for the USERID in the JCL.

NO

CA-7 will not submit a job that has a USERID in the JCL. Instead, the job is requeued to the Request queue with the status of R-NOUID at submission time.

LOADSUBC

This option can be used to suppress the submit check on the LOAD(H) command. This option only has meaning if the EXTERNAL options include SUBCHECK.

YES

This is the default. The LOAD(H) commands are validated with SUBCHECK.

NO

The LOAD(H) commands are exempted from the SUBCHECK validation.

LOGOPID

Optional. Specifies whether the transaction log records for /LOGON commands should include operator ID. In either case, password values are not logged.

YES

Indicates that transaction log records for /LOGON commands should include operator ID. This is the default.

NO

Indicates that transaction log records for /LOGON commands should not include operator ID.

PROPAGATE

Pertains only to RACF (and other SAF environments). This determines the method that CA-7 will use to associate a USERID with a job when it is submitted. This parameter is only applicable if a SUBUID hierarchy is also specified.

NO

Indicates that CA-7 will insert a USER= parameter in the jobcard when a job is submitted. This is the default.

YES

Indicates that CA-7 will not modify the JCL being submitted. Instead, the USERID will be propagated to the submitted job because the job's USERID will be used when the internal reader is opened to write the JCL. This process is similar to a job submitted through TSO inheriting the USERID of the person who submitted it.

PSOWNER

Determines whether a USERID is required to be the same as the job's OWNER in order to access a job on the CA-7/PERSONAL SCHEDULING screen.

YES

Indicates that the validation is done. The check requires that the USERID match the OWNER in order to allow access to the job. This is the default.

NO

The validation is not done.

RCLASS

Specifies the resource class being used for security calls that are made from CA-7 to validate a user's authority to access a UID Resource during CA-7 logon and when issuing the /UID,R= command. The default resource class is PANEL. (The access level is READ.)

RESLOGON

After an online terminal is logged on, subsequent LOGONs from the command line are not allowed unless RESLOGON=NO.

YES

Any /LOGON command from the top line is treated as an error requiring a logon from the formatted logon screen. This is the default.

NO

/LOGON command is allowed.

RLOGUID

Determines whether the LRLOG command (List Run Log) subjects job-related events to CA-7 UID internal security checks. See the *CA-7 Commands Guide* for more information about the LRLOG command.

YES

Indicates UID checking should be performed for LRLOG. It causes LRLOG to display only jobs that the LRLOG requestor has access to. This is the default.

NO

Indicates UID checking should not be performed for LRLOG.

STATSID

Controls disposition of USERID in PDS directory data when using the CA-7 editor. YES is the default. Members built with CA-Endevor in a CA-Endevor library do not have the USERID placed in the STATS.

YES

Indicates that the USERID is written to the PDS directory.

NO

Indicates that the USERID is not written to the PDS directory but is written out as all @s.

SUBNOID

Specifies the disposition of jobs which do not have a valid USERID available at job submission time.

NO

Indicates that jobs without a USERID cannot be submitted. Jobs without a valid USERID are moved back to the request queue and are marked with a requirement status of R-NOUID (No USERID). A requirement of R-NOUID may be satisfied in two ways. If the QJCL subparameter was selected on the SUBUID parameter, edit and replace the Queue JCL to set the USERID of the Queue JCL editor. The second method is to manually insert a USERID into the JCL from the Queue JCL Edit panel. The USERID will be identified by CA-7 and thereby satisfy the R-NOUID requirement. This is the default.

Note: For nonexecutable jobs with R-NOUID, a top line QJCL can be done and a REPLACE function. If QJCL is in the hierarchy, then the R-NOUID is satisfied. SUBUID must be coded if SUBNOID=NO is coded.

YES

Indicates that jobs may be submitted without a USERID.

SUBUID

Specifies a hierarchy of candidate USERID sources for USERID insertion during job submission. If CA-7 will be inserting USERIDs into JCL during submission, this list prioritizes the potential sources for USERIDs. The order of specification in the list determines the priority of the USERIDs to be selected.

If the SUBUID keyword is added to the SECURITY statement and CA-7 is recycled, any jobs already in the request queue are not affected. They need to be canceled and DEMANDed back in to use the new security data.

OWNER

Indicates the Job OWNER ID is to be selected for insertion into the JCL for a Job during submission.

REQ

Indicates the requester's ID is a candidate for USERID insertion. The Requester ID can be the ID of a user issuing a DEMAND, LOAD, or RUN command to request a job. The Requester ID can also be the USERID selected for a job (requester) which then triggers additional jobs. The USERID is inherited by the triggered jobs. In the case of data set triggers, jobs which create or "post" a data set to CA-7 will have their associated USERID propagated to any subsequently triggered jobs. For the U7SVC and SASSBCLP facilities, the USERID is extracted from the current environment from which the user issues the data set creation or post request.

QJCL

Indicates that the USERID of any user editing Queue JCL for a job is a candidate for USERID insertion. This would be the USERID of the last person to edit Queue JCL for a job.

CA7

Indicates the USERID assigned to CA-7 may be selected for USERID insertion. If selected, the CA-7 USERID will be inserted into the jobs JCL during job submission. For CA-ACF2, if started task checking is activated, this option may not be used.

UID

Specifies a UID Resource Table which was built using the CA7RTBL macro. If this parameter is specified, the UID Resource Table is loaded during CA-7 initialization.

USER

Optional. Specifies the name of the load module link edited from the USERID macro assembly.

XPSSID

Declares the 1 to 8 character USERID to be used in the terminal logon for XPS job submission if no USERID is supplied on the submission request from the XPS CLIENT (typically Unicenter TNG). This will be regarded as the requester ID for purposes of USERID insertion and propagation. The default value is **NONE** which means there is no default USERID, and any XPS submission requests which do not have an explicit USERID are rejected.

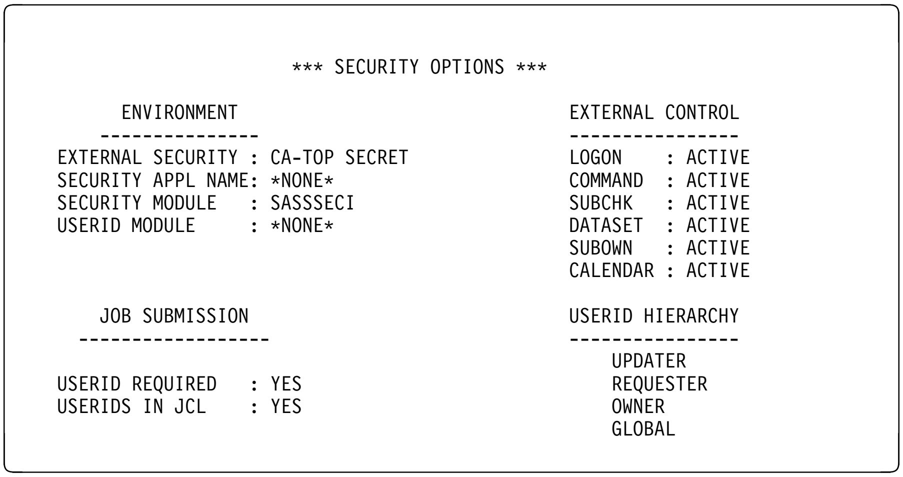
3.2 /DISPLAY,ST=

Use the /DISPLAY,ST=SEC command to display the current security options in effect for CA-7. The options displayed are based on parameters selected for the Security statement in the CA-7 initialization file. The /DISPLAY,ST= command has a STATUS keyword subparameter, SEC, which is shortened from SECURITY.

/DISPLAY,ST=

```
►►──/DISPLAY,ST=──┬─SEC──────┬──────────────────────────────────────►◄
                  └─SECURITY─┘
```

This is the /DISPLAY,ST=SEC screen.



```
                      *** SECURITY OPTIONS ***

      ENVIRONMENT                             EXTERNAL CONTROL
    ---------------                           ----------------
EXTERNAL SECURITY : CA-TOP SECRET             LOGON    : ACTIVE
SECURITY APPL NAME: *NONE*                    COMMAND  : ACTIVE
SECURITY MODULE   : SASSSECI                  SUBCHK   : ACTIVE
USERID MODULE     : *NONE*                    DATASET  : ACTIVE
                                              SUBOWN   : ACTIVE
                                              CALENDAR : ACTIVE

    JOB SUBMISSION                            USERID HIERARCHY
  ------------------                          ----------------
                                                  UPDATER
USERID REQUIRED   : YES                           REQUESTER
USERIDS IN JCL    : YES                           OWNER
                                                  GLOBAL
```

3.2.1 Fields

ENVIRONMENT
The options listed indicate the external security package present, if any, and the Security Application name for CA-7 defined to external security.

EXTERNAL SECURITY
Indicates the external security package found during CA-7 initialization. The possible values are:

CA-Top Secret
CA-ACF2
OTHER (SAF) *
NONE

* - Indicates an SAF-compatible security system, (such as, RACF).

SECURITY APPL
This is the application name for CA-7, defined to the external security package.

SECURITY MODULE
The security module built using the CA-7 SECURITY macro and identified by the NAME= parameter on the SECURITY statement in the CA-7 initialization file.

USERID MODULE
The USERID module built using the CA-7 USERID macro which defines any correspondence between various UID values under CA-7 Internal Security.

EXTERNAL CONTROL
This section displays the status of external security control for specific security functions under CA-7. The status is ACTIVE for each function that is controlled by external security and INACTIVE for those functions under the control of CA-7 internal security.

LOGON Sign on and Sign off

COMMAND
Command and Panel Security

SUBCHK Job Submission Authority

DATASET
Data Set Security

SUBOWN Job Owner ID Security

CALENDAR
CA-7 base calendar security

JOB SUBMISSION
Indicates whether a USERID is required for a job to be submitted.

USERID REQUIRED
USERID required in job.

YES USERID required. Jobs are not submitted without a USERID.

NO USERID not required. Jobs are submitted without a USERID.

USERIDS IN JCL
USERID coded in JCL.

YES USERID can be coded in JCL.

NO Jobs will not be submitted if a USERID is coded in JCL.

USERID HIERARCHY
A list of candidate USERID sources, in priority order, from which a USERID may be selected for JCL insertion during job submission.

UPDATER
The job owner USERID.

REQUESTER
The USERID of the last Queue JCL updater.

OWNER The USERID of a user requesting a job or the USERID from a triggering job.

GLOBAL The CA-7 USERID.

Chapter 4. Implementing CA-7 Security with CA-Top Secret

This chapter describes the steps necessary to implement the CA-7 External Security Interface with CA-Top Secret. A working knowledge of the security structure for CA-Top Secret and its associated command syntax is required. All of the CA-Top Secret commands shown in this chapter must be executed under CA-Top Secret.

4.1 Defining CA-7 to CA-Top Secret

The following topics provide examples of CA-Top Secret commands which can be used to implement CA-7 External Security. Refer to the *CA-Top Secret TSS Command Function Guide* for additional information on the commands listed in this chapter.

Note: The security definitions provided in this section are recommendations for establishing your CA-7 security environment. It is the responsibility of each site to determine if these recommendations meet local auditing and security standards.

4.2 Defining the CA-7 Facility

The CA-Top Secret Facility Matrix allows installations to define CA-7 as a special facility to be protected by CA-Top Secret. The facility definition is used to define specific execution requirements for CA-7 such as authorization to submit jobs, identify CA-7 as a multiuser single address space system, and to prevent abends due to single user security violations. The following command can be used to define the CA-7 facility:

Note: The definition of the CA-7 facility using the TSS modify command is not permanent. It is recommended that you add the CA-7 facility definition commands to the CA-Top Secret startup parameter file to ensure that the CA-7 facility is defined during CA-Top Secret initialization. Refer to the *CA-Top Secret Installation Guide* for a discussion of the TSSPARM0 parameter file.

4.2.1 Syntax

```
TSS MODIFY(FAC(CA7=ASUBM,MULTIUSER,NOABEND,PGM=SAS,LOG(ALL)))
```

Where:

TSS MODIFY
Specifies the CA-Top Secret command identifier (TSS). The MODIFY option must be used when referencing the CA-Top Secret Facilities Matrix.

FAC
Specifies the CA-Top Secret command parameter used to add, change, or delete facilities.

CA7
Specifies the name used in this example for the CA-7 facility.

ASUBM
Identifies the CA-7 facility as authorized for Job Submission.

MULTIUSER
Identifies the CA-7 facility as a multiuser address space.

NOABEND
Specifies that the CA-7 facility does not abend if one user in the CA-7 multiuser address space causes a security violation.

PGM=SAS
Identifies the first three characters of the program name which issues SVC calls for security validations under the CA-7 facility. Required.

LOG(ALL)
Specifies that CA-Top Secret logs all security events for the CA-7 facility.

4.3 Defining the CA-7 ACID

CA-7 requires an ACID definition to execute under CA-Top Secret security. This definition identifies CA-7 as a started task, names the procedure from which CA-7 executes, and associates the CA-7 facility with the CA-7 ACID.

4.3.1 Syntax

```
TSS CREATE(CA7ONL) NAME('CA-7 ONLINE ACID') FAC(STC,BATCH) +
    TYPE(USER) PASS(NOPW) DEPT(CA7OPS) MASTFAC(CA7) NOSUBCHK
```

Where:

TSS CREATE
Indicates the CA-Top Secret command used to create ACIDs.

CA7ONL
Indicates the name chosen in this example for the CA-7 online ACID.

NAME
Specifies the CA-Top Secret Create command parameter used to describe the ACID you are creating. In this case, the CA-7 online ACID.

FAC(STC,BATCH)
Specifies the CA-Top Secret command parameter used to define the CA-7 ACID as a started task and to allow batch job submission.

TYPE(USER)
Identifies the CA-7 ID as a User ACID.

PASS(NOPW)
Indicates that the CA-7 ACID does not require a password.

DEPT(CA7OPS)
Establishes the owning department for the CA-7 ACID.

MASTFAC(CA7)
Identifies the "master facility" for the CA-7 online ACID. This is the facility ID defined previously.

NOSUBCHK
Indicates to CA-Top Secret that CA-7 is exempted from authorization checking during job submission. This parameter is optional but recommended. Without this parameter, each USERID in JCL submitted by CA-7 must be defined to the CA-7 ACID with a PERMIT command or CA-7 abends during job submission.

The following command adds the CA-7 ACID to the CA-Top Secret Started Task facility and identifies the CA-7 procedure name.

4.3.2 Syntax

```
TSS ADDTO(STC) PROC(CA7ONL) ACID(CA7ONL)
```

Where:

TSS ADDTO(STC)
Specifies the CA-Top Secret command used to add information to the Started Task Facility.

PROC(CA7ONL)
Identifies the procedure name to be used for the CA-7 started task.

ACID(CA7ONL)
Associates the started task procedure name with the CA-7 ACID.

4.4 Defining ICOM to CA-Top Secret

The CA-7 Independent Communications Manager (ICOM) must also be defined to CA-Top Secret. ICOM is responsible for handling SMF data for jobs submitted through CA-7, and therefore requires an ACID to execute in a CA-Top Secret secured environment. The following command example may be used to define CA-7 ICOM to CA-Top Secret.

4.4.1 Syntax

```
TSS CREATE(CA7ICOM) NAME('CA-7 ICOM') FAC(STC) TYPE(USER) +
    PASS(NOPW) DEPT(CA7OPS) MASTFAC(CA7) NOSUBCHK
```

Where:

TSS CREATE
Specifies the CA-Top Secret command used to create ACIDs.

CA7ICOM
Specifies the name chosen in this example for the CA-7 ICOM ACID.

NAME
Specifies the CA-Top Secret Create command parameter used to describe the ACID you are creating. In this case, the CA-7 ICOM ACID.

FAC(STC)
Specifies the CA-Top Secret command parameter used to define the ICOM ACID as a started task.

TYPE(USER)
Identifies the CA-7 ID as a User ACID.

PASS(NOPW)
Indicates that the CA-7 ACID does not require a password.

DEPT(CA7OPS)
Establishes the owning department for the CA-7 ACID.

MASTFAC(CA7)
Identifies the master facility for the ICOM ACID. This is the facility ID defined previously.

NOSUBCHK
Indicates to CA-Top Secret that CA-7 is exempted from authorization checking during job submission. This parameter is optional but recommended. Without this parameter, each USERID in JCL submitted by CA-7 must be defined to the CA-7 ACID with a PERMIT command or CA-7 abends during job submission.

The following command adds the CA-7 ICOM ACID to the CA-Top Secret Started Task Facility and identifies the CA-7 ICOM procedure name.

4.4.2 Syntax

```
TSS ADDTO(STC) PROC(CA7ICOM) ACID(CA7ICOM)
```

Where:

TSS ADDTO(STC)
Specifies the CA-Top Secret command used to add information to the Started Task Facility.

PROC(CA7ICOM)
Identifies the procedure name to be used for the CA-7 ICOM started task.

ACID(CA7ICOM)
Associates the started task procedure name with the CA-7 ICOM ACID.

4.5 Controlling User Access to CA-7

Since CA-7 is a VTAM application with access to system resources, sign-on or logon to it must be controlled. This control is accomplished through the Facility definition previously presented, the Computer Associates Standard Security Facility (SSF), and a CA-7 initialization parameter. Refer to Chapter 3, “CA-7 Security Initialization Options” on page 3-1 for a discussion of the CA-7 initialization parameters.

With TSS controlling access to CA-7, each user must be granted authority to log on to CA-7. This authority is provided with the following CA-Top Secret command.

4.5.1 Syntax

```
TSS  ADDTO(USER)  FAC(CA7)
```

Where:

TSS
Identifies a CA-Top Secret command.

ADDTO
Specifies the CA-Top Secret command used to grant access to resources.

(USER)
Specifies the User ACID to be permitted access to the CA-7 facility.

FAC
Specifies the CA-Top Secret keyword used to identify a facility.

(CA7)
Specifies the name used in this example for the CA-7 facility as defined previously.

4.6 Defining CA-7 Command Security

CA-7 command security includes security for top line commands, panel access, and functions within a panel. All commands have a unique resource name which can be secured using the CA-Top Secret PERMIT command. This permits or authorizes users to access a given command.

The same is true for panels in CA-7. Panels have a unique panel-ID which can be specified under CA-Top Secret as a resource name to restrict access to CA-7 applications. Permitting access to a resource does not grant full functional authority for a given command or panel. Each panel may require an additional access level to have the authority for a function. The ACCESS keyword on PERMIT commands is used to grant additional authority levels for resources. The valid Access levels for CA-Top Secret are READ, CREATE, SCRATCH, UPDATE, and CONTROL.

Refer to Appendix A, “CA-7 Security Tables” on page A-1 for a list of CA-7 commands and panels and their associated resource names. The following examples illustrate the use of the CA-Top Secret PERMIT command to authorize access to CA-7 commands and panels.

Note: When defining access to command and panel resources for CA-7, the resource type must be PANEL. This is the resource type used during security calls to external security.

4.6.1 Syntax

```
TSS PERMIT(CA7USER) PANEL(L2DB1)
```

Where:

TSS PERMIT
Specifies the CA-Top Secret command used to authorize access to a resource.

CA7USER
Specifies the user ACID to be granted read access to panel resource L2DB1.

PANEL(L2DB1)
Specifies the resource type followed by the resource name to which this command applies. The L2DB1 is the resource name associated with the DB.1 panel in CA-7.

Note: The default access granted in the example PERMIT command shown above is READ.

4.6.2 Syntax

```
TSS PERMIT(CA7USER) PANEL(L2DB1) +
    ACCESS(READ,CREATE,SCRATCH,UPDATE,CONTROL)
```

Where:

TSS PERMIT
Specifies the CA-Top Secret command used to grant access to a resource.

CA7USER
Specifies the user ACID to be granted access to panel resource L2DB1.

PANEL(L2DB1)
Specifies the resource type followed by the resource name to which this command applies. The L2DB1 is the resource name associated with the DB.1 panel in CA-7.

ACCESS
Specifies the CA-Top Secret keyword used to indicate specific access to a resource.

READ
Grants read access only to the indicated resource.

CREATE
Grants creation authority to the indicated resource.

SCRATCH
Grants scratch authority to the indicated resource.

UPDATE
Grants update authority to the indicated resource.

CONTROL
Allows you to specify certain controlled accesses such as a time of day. Refer to the *CA-Top Secret TSS Command Options Guide* for a complete description of the CONTROL parameter.

4.7 Securing the /MVS Command

The /MVS command allows a CA-7 user to issue an MVS console command from a CA-7 terminal. Although such a facility may prove indispensable in certain situations, the risks associated with an indiscriminate use of the command are obvious. This section discusses security considerations regarding the use of the command. For additional information about the /MVS command, refer to the *CA-7 Commands Guide*.

The /MVS command text is sent to MVS using SVC 34. The user ID that is associated with the CA-7 address space is the user ID in control when the SVC is issued.

CA-7 does not perform any special validation to verify the terminal user's authority for the MVS command attempted. If the user is allowed to issue the /MVS command, then the specified command text is sent to MVS.

It is recommended that CA-7 command security be employed to restrict /MVS command access to a limited class of privileged users.

4.8 Controlling Job Submission Under CA-7

Depending on your security options, submit checking may be done. For these options, refer to 3.1, “SECURITY Statement” on page 3-2.

4.8.1 Syntax

```
TSS  PERMIT(USERID1) ACID(USERID2)
```

Where:

TSS
Identifies a CA-Top Secret command.

PERMIT
Specifies the CA-Top Secret command used to grant access to resources.

(USERID1)
Specifies the User ACID to be permitted submission authority for another ACID.

ACID
Specifies the CA-Top Secret keyword used to identify a USERID.

(USERID2)
Specifies the ACID for which another USER has submission authority.

4.9 Program Protection

CA-Top Secret, by default, restricts access to all programs. CA-7 requires access to numerous programs which are critical to production processing. Due to the number of modules required during execution, a program name prefix of SAS can be used when defining program access for CA-7. The main driver module UCC7 is also required. Use the following CA-Top Secret PERMIT commands to grant program access to CA-7.

4.9.1 Syntax

```
TSS PERMIT(CA7ONL) PROG(UCC7)
```

Where:

TSS
Identifies the following data as a CA-Top Secret command.

PERMIT(CA7ONL)
Specifies the CA-Top Secret keyword used to grant access to the specified resource for the CA-7 online ACID.

PROG(UCC7)
Specifies the CA-Top Secret keyword which identifies the resource to which this command applies.

UCC7
Identifies the CA-7 main driver module.

4.9.2 Syntax

```
TSS PERMIT(CA7ONL) PROG(SAS)
```

Where:

TSS
Identifies the following data as a CA-Top Secret command.

PERMIT(CA7ONL)
Specifies the CA-Top Secret keyword used to grant access to the specified resource for the CA-7 online ACID.

PROG(SAS)
Identifies a program prefix of SAS. This command grants CA-7 the authority to execute any program with a prefix of SAS.

4.9.3 Batch Users

All batch USERIDs which are associated with jobs submitted through CA-7 require access to the program SASSJJCL. This is the LOAD program which identifies resources used by a job to CA-7. To prevent the unauthorized use of CA-7, it is recommended that access to the following programs be secured:

SASSBCLP	Batch Card Load Program
SASSBSTR	Batch Terminal Interface
SASSTRLR	Trailer Step Facility
U7SVC	CA-7 SVC Facility

4.10 Job Submission

CA-7 maintains a record of USERIDs which can be associated with a job from queue entry to job submission. If requested, a USERID can be inserted into a job's JCL, prior to submission, to satisfy batch security requirements on your system. CA-7 has five potential sources for USERIDs:

Job Owner
A USERID specified in the OWNER field for a job on the DB.1 Job Definition panel. (SUBUID value of OWNER)

JCL ID This is a USERID that exists in a job's JCL at entry into the request queue. If a job's JCL contains a USERID at queue entry, USERID insertion does **not** take place. The JCL ID overrides all other USERIDs.

Requester This is the USERID of the user which requests a job through the DEMAND, LOAD, or RUN commands. (SUBUID value of REQ)

Queue JCL
This is the USERID of a user editing a job's queued JCL in the CA-7 request queue. (SUBUID value of QJCL)

CA-7 This is the USERID assigned to CA-7 at startup. If requested the CA-7 ID is propagated to submitted jobs. (SUBUID value of CA7)

The priority of USERID sources is determined by the specification of the SUBUID keyword on the SECURITY statement in the CA-7 initialization file. The SUBUID keyword specifies a hierarchy of USERIDs for JCL insertion.

At submission time, CA-7 scans the USERID hierarchy to determine if a USERID is available from the first hierarchy entry. If an ID is found, it is inserted into the job's JCL and the job is submitted, assuming all other requirements have been met. If an ID was not found, the next source entry is checked for an available ID.

This process continues until an ID is found and inserted into the JCL. If all potential sources have been checked and a USERID is not available, CA-7 checks the status of the SUBNOID flag. The SUBNOID flag is set by the SUBNOID keyword specified on the SECURITY statement in the CA-7 initialization file. If SUBNOID=YES, a job may be submitted without a USERID. If SUBNOID=NO, jobs may not be submitted without a valid USERID and are moved back to the request queue with a requirement status of R-NOUID. The R-NOUID status indicates that all USERID sources were checked and no valid USERID was found for JCL insertion.

4.10.1 Satisfying the R-NOUID Requirement

There are two ways to satisfy the R-NOUID requirement.

- If you have specified the QJCL keyword on the SUBUID parameter, you can FETCH/EDIT the job's queued JCL and immediately do a SAVE/REPLACE. CA-7 saves the USERID of the user editing the queued JCL. This would satisfy the R-NOUID requirement because an ID is now available from one of the candidate USERID sources and the job is now eligible for submission.
- The second method is to manually add a USERID to the job's queued JCL. CA-7 recognizes the addition of the USERID to the JCL and the R-NOUID requirement would be satisfied.

Note: If a job's JCL contains a USERID at queue entry time, this USERID overrides all other USERIDs and USERID insertion will not take place.

You may not manually satisfy or POST an R-NOUID requirement. If you try to satisfy the requirement by any method other than those listed above, the request is ignored.

4.10.2 USERID Propagation

Establishing USERIDs to be associated with jobs when submitted by CA-7 is a critical aspect of security. Defining the USERID hierarchy requires careful planning to ensure that each job is submitted with the correct ID and therefore the proper security. If Requester (REQ) is specified in the SUBUID hierarchy, USERIDs are propagated to any jobs triggered by the original request. This means that USERID propagation of a requesting ID occurs for the following conditions:

- The USERID of a user requesting work through the DEMAND, LOAD, or RUN commands.
- The USERID associated with a job which triggers any additional jobs.
- For data set triggers, the USERID associated with a job which was submitted by CA-7 and created the data set.
- For data set triggers which are initiated through U7SVC and SASSBCLP, the USERID associated with the user posting the creation of the data set.

4.10.3 JCL USERID Format

For USERID insertion, CA-7 modifies the last statement of the JCL job statement to add the USERID. A comma is added to the last statement to indicate continuation and is followed by a USER= statement to supply the USERID. The following example illustrates the JCL statement format used during ID insertion.

```
// USER=userid
```

Note: The USERID password is inserted, if required, by CA-Top Secret. This is an automatic function related to Job Submission and the ASUBM keyword specified on the CA-7 facility which was previously defined.

4.11 UID Resources

Access to information on the CA-7 database is controlled through UID security validation. When a user attempts to access a job on the database, regardless of whether internal or external security is in control, the user's UID value is checked against the UID value associated with the job. This provides job-level security for the CA-7 database. External security does not provide an equivalent JOB level protection; therefore, it is important that each CA-7 user be assigned a UID which relates to the user's area of responsibility.

Note: UID resource security is only valid in an environment where external security controls CA-7 logons. Calls are made to the external security package to validate a user's authority to access the resource. The resources have no meaning to CA-7 internal security.

The UID value (0-255) can be obtained from the USERID entry in the CA-7 internal security module, through UID resource validation during logons to CA-7 or through the /UID top line command issued under a CA-7 session. If you wish to maintain USERIDs in the CA-7 internal security module, refer to Chapter 7, “CA-7 Internal Security” on page 7-1 for details on defining USERIDs in the internal security module. The following information outlines the steps necessary to implement UID resource validation and describes the security processing involved.

UID resource security requires a UID Resource Table which CA-7 references during UID resource validation. This table contains resource names and associated UID value entries to be used during the UID validation process. A sample resource table, SASSRTBL, is provided in both source and load module format and may be used to implement UID resource security. To create a site specific UID Resource Table with unique resource names and UID values, use the CA7RTBL macro to generate the table.

Note: The default resource class for UID resources is PANEL. You may change the resource class for calls to external security using the RCLASS= parameter on the CA-7 initialization file SECURITY statement.

You may use the /PROF command to establish and maintain a default UID resource for users logging on to CA-7. See the *CA-7 Commands Guide* for a description of the /PROF command.

4.11.1 CA7RTBL Macro

The CA7RTBL macro is used to generate the UID Resource Table. Refer to the description below on the required parameters for the macro. Once the new source has been created, refer to member UL23309 in the CA-7 SAMPJCL file for applying the USERMOD.

The following is an example of the CA7RTBL macro statement.

```
CA7RTBL RSRC=CA70255,UID=255
```

Where:

CA7RTBL
This is the UID Resource Table generation macro. It is used to build the UID Resource Table.

RSRC
The resource name to be generated in this entry of the table. The resource name can be a 1- to 8-character name which meets site specific external security resource naming conventions.

Note: Resource names must not conflict with existing panel or command names if the default PANEL resource class is used. See the description of the RCLASS keyword on the SECURITY statement in Chapter 3.

UID
The value which will be associated with the resource name supplied on the RSRC= parameter. The value can be from 0 (zero) to 255.

4.11.1.1 Usage Notes

1. The UID Resource Table name must be a valid PDS member name.
2. The CA7RTBL macro must be coded starting in column 10.
3. Duplicate resource names are not allowed, but duplicate UID values are allowed in the table.
4. The UID Resource Table source must be assembled and link-edited into a load library accessible by CA-7.
5. The last entry in the table must be specified with a resource name of LAST (RSRC=LAST) to indicate the end of the table. The UID= parameter is not necessary on the last statement.
6. The UID Resource Table name must be identified on the CA-7 initialization file SECURITY statement using the UID= parameter. Refer to 3.1, “SECURITY Statement” on page 3-2 for a discussion of the SECURITY statement and its associated parameters.
7. The resource names coded in the UID Resource Table must be defined to external security.

4.11.1.2 UID Resource Table - SASSRTBL Source

```
Example UID Resource Table - SASSRTBL Source

         TITLE 'CA-7 EXTERNAL SECURITY UID/RESOURCE TABLE'              00010008
SASSRTBL START 0                                                        00020000
SASSRTBL CA7RTBL RSRC=CA70000,UID=000                                   00040008
         CA7RTBL RSRC=CA70001,UID=001                                   00050008
         CA7RTBL RSRC=CA70002,UID=002                                   00070008
         CA7RTBL RSRC=CA70003,UID=003                                   00090008
         CA7RTBL RSRC=CA70004,UID=004                                   00101008
         CA7RTBL RSRC=CA70005,UID=005                                   00103008
         CA7RTBL RSRC=CA70006,UID=006                                   00105008
         CA7RTBL RSRC=CA70007,UID=007                                   00107008
         CA7RTBL RSRC=CA70008,UID=008                                   00109008
         CA7RTBL RSRC=CA70009,UID=009                                   00109208
         CA7RTBL RSRC=CA70010,UID=010                                   00109408
                    .                                                   00109608
                    .                                                   00109808
                    .                                                   00110008
         CA7RTBL RSRC=CA70254,UID=254                                   00142808
         CA7RTBL RSRC=CA70255,UID=255                                   00142908
         CA7RTBL RSRC=LAST                                              00143008
         END                                                            00150000
```

4.11.2 CA-7 Logon and UID Resource Validation

When a user signs on to CA-7, the user can optionally supply a UID resource name in the UID RESOURCE field on the CA-7 Logon screen. If no UID resource name is supplied, a check is made to see if one is defined in the user's CA-7 profile record. If present, the profile resource name is used as if it were entered on the Logon screen. The USERID and PASSWORD supplied are first validated through external security and then a lookup is performed to determine if the user is defined in the CA-7 Internal Security module.

If the user is defined in the Internal Security module, any UID resource name passed on the Logon screen is ignored. If the user is not defined in the Internal Security module, the UID Resource Table is searched to find a matching resource entry.

If the resource name is not found in the UID Resource Table, the user is signed on to CA-7 with a UID value of 0 (zero). If a matching resource entry is found, a call is made to external security to validate the user's authority to access the resource.

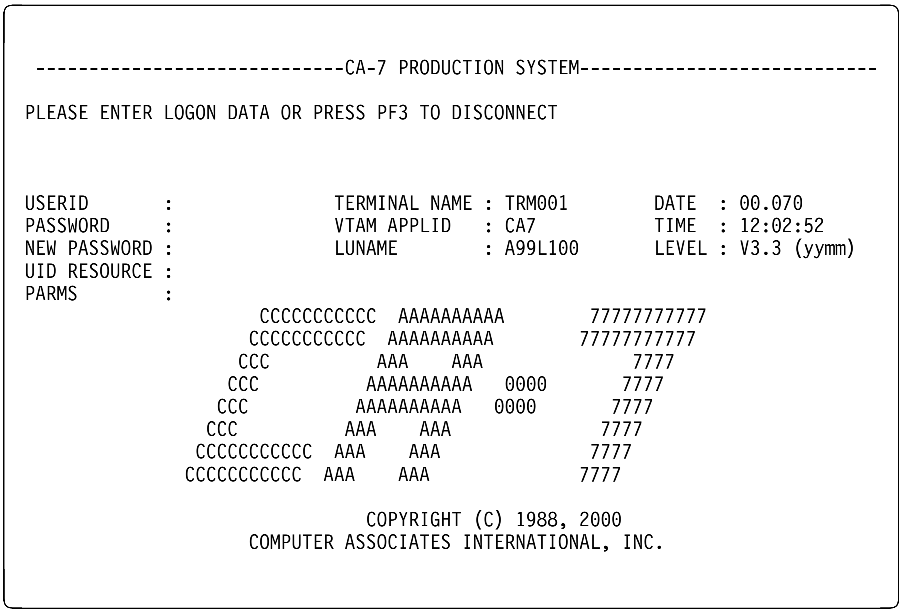
If the user is not authorized to access the resource, a message is displayed indicating the failure and the logon attempt fails. If the user is authorized to access the resource, the associated UID value in the UID Resource Table is assigned to the user.

This process allows external security to control UID assignment to CA-7 users and eliminates the need to maintain all USERIDs in the CA-7 Internal Security module.

Note: The UID resource validation process is the same under the CA-7 ISPF Interface; however, no password is required.

The following is a sample CA-7 Logon screen.

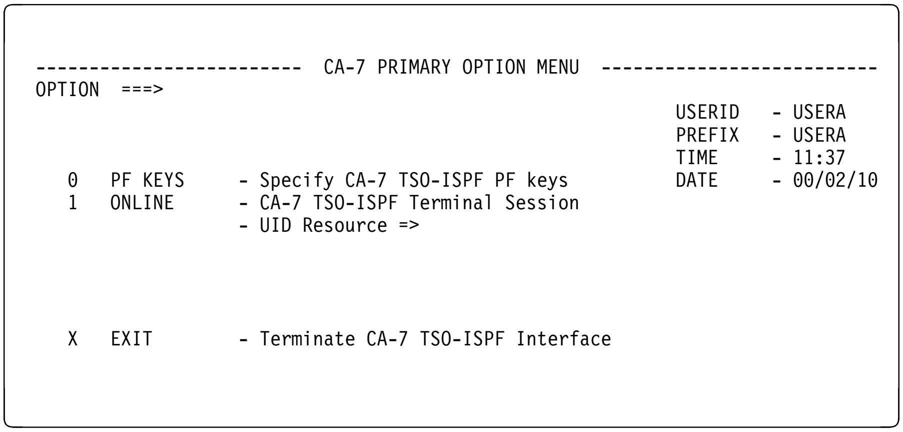
CA-7 Logon Screen



```
 -----------------------------CA-7 PRODUCTION SYSTEM----------------------------

PLEASE ENTER LOGON DATA OR PRESS PF3 TO DISCONNECT



USERID       :               TERMINAL NAME : TRM001        DATE  : 00.070
PASSWORD     :               VTAM APPLID   : CA7           TIME  : 12:02:52
NEW PASSWORD :               LUNAME        : A99L100       LEVEL : V3.3 (yymm)
UID RESOURCE :
PARMS        :
                      CCCCCCCCCCC  AAAAAAAAAA        77777777777
                     CCCCCCCCCCC  AAAAAAAAAA        77777777777
                    CCC          AAA    AAA              7777
                   CCC          AAAAAAAAAA   0000       7777
                  CCC          AAAAAAAAAA   0000       7777
                 CCC          AAA    AAA              7777
                CCCCCCCCCCC  AAA    AAA              7777
               CCCCCCCCCCC  AAA    AAA              7777

                                COPYRIGHT (C) 1988, 2000
                     COMPUTER ASSOCIATES INTERNATIONAL, INC.
```

The following is a sample CA-7 ISPF Interface Primary Option Menu screen.

CA-7 ISPF Interface Primary Option Menu



```
-------------------------  CA-7 PRIMARY OPTION MENU  --------------------------
OPTION  ===>
                                                            USERID   - USERA
                                                            PREFIX   - USERA
                                                            TIME     - 11:37
   0   PF KEYS     - Specify CA-7 TSO-ISPF PF keys          DATE     - 00/02/10
   1   ONLINE      - CA-7 TSO-ISPF Terminal Session
                   - UID Resource =>




   X   EXIT        - Terminate CA-7 TSO-ISPF Interface
```

Enter the END command to terminate CA-7 TSO-ISPF.

4.12 /UID Command

The /UID command is used to change a user's current UID processing value through UID resource validation. The /UID command requires that the UID Resource Table option be implemented and that the resources and the appropriate security authorization are defined to external security.

4.12.1 Example

```
     /UID,R=XXXXXXXX
     /UID,LIST
```

Where:

R=
Identifies a resource name which exists in the UID Resource Table. The user's authorization to access the resource is validated through external security and if authorized, the user's current UID value is updated to reflect the UID value found in the UID Resource Table associated with the resource name that was supplied.

LIST
Displays all resource entries and the associated UID values in the UID Resource Table.

4.13 /REFRESH Command

The /REFRESH command is used to refresh the UID Resource Table that was loaded during CA-7 initalization without cycling CA-7. The definitions coded within the specified module will completely replace the definitions currently being used. However, any changes made in the actual CA-Top Secret definitions (using CA-Top Secret commands) may not take effect until CA-7 is recycled.

```
    /REFRESH,MOD=xxxxxxxx
```

Where:

MOD=

Identifies a UID Resource Table in load module format that was built using the CA7RTBL macro.

xxxxxxxx

This must be the member name of the UID Resource Table, and it must reside in a load library accessible to CA-7.

4.14 External Communicators with CA-Top Secret

The external communicators (SASSBSTR, SASSTRLR, U7SVC, and SASSBCLP) provide a means for users outside the CA-7 address space to communicate with CA-7. Because the use of these programs may allow access to production jobs, it is recommended that careful consideration be given to the question of access to these facilities. Users of the Batch Terminal Interface require access to the program SASSBSTR. Users of the Trailer step, the Batch Card Load Program and U7SVC must be given access to SASSTRLR, SASSBCLP, and U7SVC respectively. Once the question of program access is settled, additional controls may be implemented to prevent unauthorized use of these facilities. This section describes those controls.

Two types of communication with CA-7 are supported with the external communicators:

- Terminal communication (Batch Terminal Interface, Trailer and U7SVC)
- Data set posting (U7SVC and SASSBCLP)

4.14.1 Terminal Communication

The Batch Terminal Interface (SASSBSTR), the Trailer facility (SASSTRLR), and U7SVC each allow the user to send terminal commands to CA-7. Although no online terminal is used with this mode of communication, input from these programs is treated as terminal input by CA-7. Command security in these environments is handled as it is for all CA-7 terminals. CA-Top Secret controls access to CA-7 commands if EXTERNAL=COMMAND is specified on the SECURITY statement in the CA-7 initialization file. CA-Top Secret determines a user's access to CA-7 terminal commands based on the USERID supplied on the /LOGON command. Thus when using an External Communicator, any command input must be preceded by a /LOGON command.

CA-Top Secret normally requires a password at logon. But including passwords in command input for the External Communicators would obviously represent a serious security exposure. Several checks may be made to avoid the need to include passwords in command input when using these facilities. If no /LOGON command is found in the command input, then a /LOGON statement is built using the USERID associated with the current user. Under certain conditions it may not be possible to extract the USERID associated with the user of the External Communicator. In that event, a /LOGON statement is built using a default USERID of CA7DUMMY. If a /LOGON statement is found in the command input, the current user's authority to use the USERID found on the /LOGON statement may be checked. If the USERID found on the /LOGON statement matches the USERID of the current user, it is assumed that the user has the authority to use the USERID. If the USERIDs differ, a check may be made to validate the user's READ access to an entity whose name is the USERID found on the /LOGON statement. The CA-Top Secret PERMIT command can be used to define this relationship in the same way shown in 4.8, “Controlling Job Submission Under CA-7” on page 4-13. If a /LOGON statement was generated or if the user's authority to use a USERID was successfully validated, CA-7 allows the user to LOGON without a password.

Note: The USERID of the current user is determined by using CAS9 SSF services. Refer to the Unicenter TNG Framework for OS/390 documentation for more information about SSF.

Submit checking for External Communicators may be activated by modifying ICMDSECT. Refer to the UL233IZ member of SAMPJCL for more information.

4.14.2 SASSTRLR and External Security

The following information is intended to show the actions that are performed by SASSTRLR during execution with relation to security.

A security EXTRACT is done to determine the USERID of the submitted job. This USERID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied or if a logon statement was supplied without an OPID, one is generated for the execution. The logon statement resembles the following:

```
/LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

4.14.3 SASSBSTR and External Security

The following information is intended to show the actions that are performed by SASSBSTR during execution with relation to security.

A security EXTRACT is done to determine the USERID of the submitted job. This USERID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied, then one is generated for the execution. The logon statement resembles the following:

```
  /LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

4.14.4 U7SVC and External Security

The following information is intended to show the actions that are performed by U7SVC during execution with relation to security. There are two different paths that can be taken. It depends on whether there is an input stream with the U7SVC or if it is just a D= to post a data set.

4.14.4.1 U7SVC with D= PARM

A security EXTRACT is done to determine the USERID that invoked U7SVC. This USERID is later used to generate a full logon statement or optionally perform a data set create check.

If the SVDSNCHK bit is not turned on, the D= command is passed through to CA-7 with no further security checking to be done by U7SVC.

If the SVDSNCHK bit is turned on, then a security call is made by U7SVC. This call determines if the EXTRACTed ID has CREATE authorization for the data set specified on the D=. If the EXTRACTed ID does have authorization, the D= command is passed to CA-7 for processing.

4.14.4.2 U7SVC with an Input Stream

A security EXTRACT is done to determine the USERID that invoked U7SVC. This USERID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied or if a logon statement was supplied without an OPID, then one is generated for the execution. The logon statement resembles the following:

```
/LOGON extid                                *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

4.14.5 Data Set Posting

The Batch Card Load Program (SASSBCLP) and U7SVC allow the user to post the creation of a data set to CA-7. Because such posting may satisfy requirements or cause job triggering, the need to secure the use of these facilities is critical. There are two features of these facilities that should be mentioned in this connection: data set access validation and USERID propagation.

The USERID associated with the user of U7SVC or SASSBCLP is extracted to determine the user's authority to create the data set. Under certain conditions it may not be possible to extract the USERID for the user of the External Communicator. In that event, a default USERID of CA7DUMMY is used.

If REQ is specified in the SUBUID hierarchy on the SECURITY statement in the CA-7 initialization file, then the USERID associated with the data set creation may be propagated to triggered jobs.

For example, suppose that a user whose USERID is XXX submits a batch job which uses U7SVC to post the creation of data set A.B to CA-7. Suppose also that the creation of this data set triggers job Z. Further suppose that REQ is in the first position in the SUBUID hierarchy. In such a case, USERID XXX could be propagated to job Z when the job is submitted.

Note: Each of the External Communicators attempts to extract the USERID of the current user, SASSBCLP and U7SVC may be made to verify the authority of the user to create that data set whose creation is to be posted to CA-7. Refer to the UL233IZ member of SAMPJCL for information on ICMDSECT modifications to implement this checking.

4.15 Sample Definitions

The member TSSSAMP in file 9 (CAI.CA7.SAMPJCL) on the installation tape contains sample TSS commands which can be used to secure the CA-7 processing environment under CA-Top Secret. The definitions are intended as examples only and should be reviewed and modified to meet your installations security requirements. Once tailored to your site's specifications, the definitions may be used as batch input to CA-Top Secret. Refer to the *CA-Top Secret Commands Function Guide* for a discussion on executing CA-Top Secret commands in batch.

Chapter 5. Implementing CA-7 Security with CA-ACF2

This chapter describes the steps necessary to implement the CA-7 External Security Interface with CA-ACF2. A working knowledge of the security structure for CA-ACF2 and its associated command syntax is required. All of the CA-ACF2 commands shown in this chapter must be executed under CA-ACF2.

Note: The security definitions provided in this section are recommendations for establishing your CA-7 security environment. It is the responsibility of each site to determine if these recommendations meet local auditing and security standards.

5.1 Defining CA-7 to CA-ACF2

CA-7 requires a LOGONID defined to CA-ACF2 to establish access authority to data sets and to allow for job submission. The LOGONID must be defined as a started task using the procedure name which invokes CA-7. Since multiple users can be logged on to CA-7 at the same time, it is referred to as a MUSASS - Multi-User Single Address Space Subsystem and requires the MUSASS attribute when defining the LOGONID.

CA-7 also requires the authorization to submit jobs. This can be accomplished by adding the SUBAUTH and JOBFROM privileges to the CA-7 LOGONID. This exempts CA-7 Submit Validation and provides uninterrupted production scheduling. The CA-ACF2 command example shown below can be used to define the CA-7 LOGONID to CA-ACF2.

Note: This command must be entered under CA-ACF2. You may use online or batch CA-ACF2 processing to define the CA-7 LOGONID.

```
INSERT  CA7  NAME(CA-7 Production ID)  STC  MUSASS  SUBAUTH  RESTRICT
DUMPAUTH  JOBFROM  NO-SAF
```

Figure 5-1. Example CA-ACF2 Command used to define the CA-7 LOGONID

Where:

INSERT
Specifies the CA-ACF2 command used to add a logonid to the CA-ACF2 database.

CA7
Specifies the name chosen in this example for the CA-7 LOGONID.

NAME
Indicates a comment field used to describe the LOGONID being added.

STC
Identifies the CA-7 LOGONID as a started task ID.

MUSASS
Defines the CA-7 LOGONID as a Multi-User Single Address Space Subsystem.

SUBAUTH
Specifies that the CA-7 LOGONID may only be submitted through APF authorized programs. This is used in combination with the RESTRICT privilege.

RESTRICT
Indicates that the LOGONID does not require a password.

DUMPAUTH
Permits the CA-7 LOGONID to generate complete dumps of its address space in the event of an abend.

JOBFROM
Allows CA-7 to use the JOBFROM statement for USERID insertion at job submission time.

NO-SAF
Indicates that all calls made to the System Authorization Facility (SAF) on behalf of the user are ignored by the CA-ACF2 SAF Interface.

Note: If you already have a started task LOGONID with similar privileges and attributes, you may use the INSERT USING command option when defining the CA-7 LOGONID. Refer to the *CA-ACF2 MVS Administrator Guide* for a complete description of the USING command option and the additional privileges and attributes shown in this example.

5.2 Defining CA-7 as a Resource

CA-7 may be defined to CA-ACF2 as a resource to control LOGON access. The resource definition for CA-7 under CA-ACF2 is not required; however, it does provide an additional level of security for restricting access to CA-7. If LOGON security for CA-7 is to be controlled through CA-ACF2, a resource check is to be made during LOGON to determine if the user is authorized to access CA-7.

The steps required to implement the application resource checking for LOGONs to CA-7 are as follows:

1. Define a Resource Rule under CA-ACF2 identifying CA-7 as a resource. If you are using CA-ACF2 6.0 or higher, you should define a CLASMAP for APPL:

```
CLASMAP.CA7 RESOURCE(APPL) RSRCTYPE(APP)
```

2. Compile and store the resource rule under CA-ACF2.
3. Add the APPL= keyword to the SECURITY statement in the CA-7 initialization file and specify the resource name.
4. Add the LOGON keyword to the EXTERNAL= parameter list on the SECURITY statement in the CA-7 initialization file.

```
  $KEY(CA7) TYPE(APP)
  *.........allow access                  comment statement
   UID(local UID string) ALLOW
  *.........disallow access               comment statement
   UID(local UID string) PREVENT
  *
```

Figure 5-2. Example CA-ACF2 Application Resource Rule for CA-7

Where:

$KEY(CA7)
Specifies the CA-ACF2 keyword used to name the resource to be protected. CA7 is the name used in this example and must match the APPL= value on the SECURITY statement.

TYPE
Identifies the type of resource rule. (APP = Application resource type)

UID
Identifies the UID string of users to be allowed or prevented from accessing this resource.

ALLOW
Specifies the CA-ACF2 keyword used to grant access to a resource.

PREVENT
Specifies the CA-ACF2 keyword used to deny access to a resource.

Note: The Application Resource rule does not take effect until the rule is compiled and stored under CA-ACF2. Refer to the *CA-ACF2 MVS Administrators Guide* for details on compiling and storing rules.

5.3 Defining CA-7 Command Security

Implementation of CA-7 command security includes securing top line commands, application panel access and panel functions. Each CA-7 panel has a unique panel-ID which can be defined as a resource to CA-ACF2. For subsequent functions on each panel, which may involve multiple access types (READ, ADD, UPDATE, and DELETE), a service level can be specified on the resource rule to provide an additional level of protection.

For example, a user enters the DB top line command to access the Database Maintenance Menu. CA-7 first checks the user's authority to access the Database Maintenance Menu (panel-ID = L2DB). If the user has the authorization, the panel is displayed. The user now selects option 1 - Job Definition from the Database Maintenance Menu. This equates to a panel-ID of L2DB1. If the user has the appropriate authority, the panel is displayed. The user now enters the LIST option from the Job Definition panel to list JOBA. This requires a service level of READ on panel L2DB1 to perform the LIST command. If the user has the required authorization, JOBA is listed. The user now attempts the UPD option to update JOBA on the CA-7 database. This requires a service level of UPDATE for panel L2DB1 to perform the update. If the user has the proper authority, the job is updated.

Remember that protection is provided not only for panels within CA-7 but for the additional functions on each panel. Each command requires a service level entry on the resource rule definition to perform that function. Refer to Appendix A, “CA-7 Security Tables” on page A-1 for a list of the CA-7 panel-IDs, commands, and access level requirements.

5.3.1 Resource Rule Masking

CA-ACF2 provides the ability to "mask" resource names to simplify the specification of access rules for a group of users. To mask a resource rule, you must enter asterisks for each level of access you wish to generically specify for users.

For example, to allow users access to all panels within the CA-7 Database Maintenance Application, you could define the resource key name as L2DB****. The asterisks mask the panel-IDs to L2DBxxxx which would include any panel-ID that has a DB prefix. This does not authorize the users to perform all functions from each panel for Database Maintenance. Authority for each function must be specified by a service level on the resource rule.

Note: CA-ACF2 requires the use of Resident Directories to use Resource Rule masking. Resident Directories may also be required for other CA-ACF2 options. Refer to the *CA-ACF2 MVS Administrators Guide* for a discussion of Resource Rule Resident Directories.

```
$KEY(L2DB1) TYPE(PAN)
*
 UID(Local UID string)  SERVICE(READ,ADD,UPDATE,DELETE)  ALLOW
*
* The above rule allows users with matching UID strings to
* access the Database Maintenance - Job Definition
* panel (L2DB1) with full function authority.
*
 UID(Local UID string)  SERVICE(READ)  ALLOW
*
* The above rule allows users with matching UID strings to
* access the Database Maintenance - Job Definition panel
* (L2DB1) with READ access authority only.
*
 UID(Local UID string) PREVENT
*
* The above rule prevents access to the Database Maintenance -
* Job Definition panel (L2DB1) for users with a matching UID
* string.
```

Figure 5-3. Example CA-7 Panel Resource Rule

Where:

$KEY(L2DB1)
Identifies the Database Maintenance - Job Definition panel. The L2 preceding the DB1 is the CA-7 product code and is required.

TYPE(PAN)
Identifies the type of resource rule.

UID
Identifies the UID string of users for which this resource rule applies.

ALLOW
Allows users with a matching UID string access to the indicated resource.

PREVENT
Prevents users with a matching UID string access to the indicated resource.

SERVICE
Specifies authority for service level access to functions on each panel. Access to a panel does not grant full access to the functions contained on that panel. The valid service levels are READ, ADD, UPDATE, and DELETE.

Note: All CA-7 panel and command resource rules under CA-ACF2 require a resource type of PAN.

```
*
$KEY(L2DB****) TYPE(PAN)
*
 UID(Local UID string) SERVICE(READ) ALLOW
*
* The rule above uses Resource Rule "masking." The Resource name
* has been "masked" using asterisks.  This rule would allow any users
* with a matching UID string access to all CA-7 Database Maintenance
* panels with a service level of READ.
*
```

Figure 5-4. Example CA-7 "Masked" Panel Resource Rule

Where:

$KEY(L2DB**)**
Identifies any CA-7 Database Maintenance panel by using Resource Rule masking. The asterisks mask the last four characters of the resource name allowing access to any panel with a prefix of L2DB.

TYPE(PAN)
Identifies the UID string of users for which this resource applies.

SERVICE(READ)
Identifies the level of access to this resource.

ALLOW
Specifies the CA-ACF2 keyword used to grant access to this resource.

Note: CA-ACF2 requires the use of Resident Directories to use Resource Rule masking. Resident Directories may also be required for other CA-ACF2 options. Refer to the *CA-ACF2 MVS Administrators Guide* for a discussion of Resource Rule Resident Directories.

5.4 Securing the /MVS Command

The /MVS command allows a CA-7 user to issue an MVS console command from a CA-7 terminal. Although such a facility may prove indispensable in certain situations, the risks associated with an indiscriminate use of the command are obvious. This section discusses security considerations regarding the use of the command. For additional information about the /MVS command, refer to the *CA-7 Commands Guide*.

The /MVS command text is sent to MVS using SVC 34. The LOGONID that is associated with the CA-7 address space is the LOGONID in control when the SVC is issued.

CA-7 does not perform any special validation to verify the terminal user's authority for the MVS command attempted. If the user is allowed to issue the /MVS command, then the specified command text will be sent to MVS.

It is recommended that CA-7 command security be employed to restrict /MVS command access to a limited class of privileged users.

5.5 Defining SUBMIT Resource Rules

To prevent unauthorized access to LOGONIDs under CA-7, you can define SUBMIT resource rules to CA-ACF2 to restrict the ability of users to access LOGONIDs other than their own. Generally LOGONIDs that are associated with a given user have established access authority which restricts their access to specific areas of responsibility. The following CA-ACF2 commands can be used to define a SUBMIT resource rule under CA-ACF2 for a LOGONID to be used under CA-7.

```
 $KEY(CA7USER) TYPE(SUB)
*
  UID(Local UID string) ALLOW
*
* The above rule allows users with matching UID strings access
* to the LOGONID CA7USER.
*
  UID(Local UID string) PREVENT
*
* The above rule disallows users with matching UID strings access
* to the LOGONID CA7USER.
*
```

Figure 5-5. Example CA-7 SUBMIT Resource Rule

Where:

$KEY(CA7USER)
Identifies the Logonid, used in this example, for which this SUBMIT resource rule applies.

TYPE(SUB)
Identifies the resource rule type. In this case SUB for SUBMIT.

UID
Identifies the UID string for which this resource rule applies.

This example illustrates giving SUBMIT authority from one USERID to another.

```
*
$KEY(USERID2) TYPE(SUB)
*
 UID(Local UID string) SERVICE(READ) ALLOW
*
*
```

Figure 5-6. Example Giving SUBMIT Authority from One USERID to Another

Where:

$KEY(USERID2)
Identifies the USERID for which the local UID string has submit authority.

TYPE(SUB)
Identifies as submit authority.

UIDSTRING
Identifies the UID string of users for which this resource applies.

SERVICE(READ)
Identifies the level of access to this resource.

ALLOW
Specifies the CA-ACF2 keyword used to grant access to this resource.

5.6 Job Submission

CA-7 maintains a record of USERIDs which can be associated with a job from queue entry to job submission. If requested, a USERID can be inserted into a jobs JCL, prior to submission, to satisfy batch security requirements on your system. CA-7 has five potential sources for USERIDs:

Job Owner
A USERID specified in the OWNER field for a job on the DB.1 Job Definition panel. (SUBUID value of OWNER)

JCL ID This is a USERID that exists in a job's JCL at entry into the request queue. If a job's JCL contains a USERID at queue entry, USERID insertion does **not** take place. The JCL ID overrides all other USERIDs.

Requester This is the USERID of the user which requests a job through the DEMAND, LOAD, or RUN commands. (SUBUID value of REQ)

Queue JCL
This is the USERID of a user editing a job's queued JCL in the CA-7 request queue. (SUBUID value of QJCL)

CA-7 This is the USERID assigned to CA-7 at startup. If requested the CA-7 ID is propagated to submitted jobs. If started task checking is activated, this option should not be used. (SUBUID value of CA7)

The priority of USERID sources is determined by the specification of the SUBUID keyword on the SECURITY statement in the CA-7 initialization file. The SUBUID keyword specifies a hierarchy of USERIDs for JCL insertion.

At submission time, CA-7 scans the USERID hierarchy to determine if a USERID is available from the first hierarchy entry. If an ID is found, it is inserted into the job's JCL and the job is submitted, assuming all other requirements have been met. If an ID was not found, the next source entry is checked for an available ID.

This process continues until an ID is found and inserted into the JCL. If all potential sources have been checked and a USERID is not available, CA-7 checks the status of the SUBNOID flag. The SUBNOID flag is set by the SUBNOID keyword specified on the SECURITY statement in the CA-7 initialization file. If SUBNOID=YES, a job may be submitted without a USERID. If SUBNOID=NO, jobs may not be submitted without a valid USERID and are moved back to the request queue with a requirement status of R-NOUID. The R-NOUID status indicates that all USERID sources were checked and no valid USERID was found for JCL insertion.

5.6.1 Satisfying the R-NOUID Requirement

There are two ways to satisfy the R-NOUID requirement.

- If you have specified the QJCL keyword on the SUBUID parameter, you can FETCH/EDIT the job's queued JCL and immediately do a SAVE/REPLACE. CA-7 saves the USERID of the user editing the queued JCL. This would satisfy the R-NOUID requirement because an ID is now available from one of the candidate USERID sources and the job is now eligible for submission.
- The second method is to manually add a USERID to the job's queued JCL. CA-7 recognizes the addition of the USERID to the JCL and the R-NOUID requirement would be satisfied.

Note: If a job's JCL contains a USERID at queue entry time, this USERID overrides all other USERIDs and USERID insertion will not take place.

You may not manually satisfy or POST an R-NOUID requirement. If you try to satisfy the requirement by any method other than those listed above, the request is ignored.

5.6.2 USERID Propagation

Establishing USERIDs to be associated with jobs when submitted by CA-7 is a critical aspect of security. Defining the USERID hierarchy requires careful planning to ensure that each job is submitted with the correct ID and therefore the proper security. If Requester (REQ) is specified in the SUBUID hierarchy, USERIDs are propagated to any jobs triggered by the original request. This means that USERID propagation of a requesting ID occurs for the following conditions:

- The USERID of a user requesting work through the DEMAND, LOAD or RUN commands.
- The USERID associated with a job which triggers any additional jobs.
- For data set triggers, the USERID associated with a job which was submitted by CA-7 and created the data set.
- For data set triggers which are initiated through U7SVC and SASSBCLP, the USERID associated with the user posting the creation of the data set.

5.6.3 JCL USERID Format

For USERID insertion, CA-7 inserts a JOBFROM statement immediately following the Job statement. The following example illustrates the JCL statement format used during ID insertion for CA-ACF2.

```
//*JOBFROM userid
```

5.7 UID Resources

Access to information on the CA-7 database is controlled through UID security validation. When a user attempts to access a job on the database, regardless of whether internal or external security is in control, the user's UID value is checked against the UID value associated with the job. This provides job-level security for the CA-7 database. External security does not provide an equivalent JOB level protection; therefore, it is important that each CA-7 user be assigned a UID which relates to the user's area of responsibility.

Note: UID resource security is only valid in an environment where external security controls CA-7 logons. Calls are made to the external security package to validate a user's authority to access the resource. The resources have no meaning to CA-7 internal security.

The UID value (0-255) can be obtained from the USERID entry in the CA-7 internal security module, through UID resource validation during logons to CA-7 or through the /UID top line command issued under a CA-7 session. If you wish to maintain USERIDs in the CA-7 internal security module, refer to Chapter 7, “CA-7 Internal Security” on page 7-1 for details on defining USERIDs in the internal security module. The following information outlines the steps necessary to implement UID resource validation and describes the security processing involved.

UID resource security requires a UID Resource Table which CA-7 references during UID resource validation. This table contains resource names and associated UID value entries to be used during the UID validation process. A sample resource table, SASSRTBL, is provided in both source and load module format and may be used to implement UID resource security. To create a site specific UID Resource Table with unique resource names and UID values, use the CA7RTBL macro to generate the table.

Note: The default resource class for UID resources is PAN. You may change the resource class for calls to external security using the RCLASS= parameter on the CA-7 initialization file SECURITY statement.

You may use the /PROF command to establish and maintain a default UID resource for users logging on to CA-7. Refer to the *CA-7 Commands Guide* for a description of the /PROF command.

5.7.1 CA7RTBL Macro

The CA7RTBL macro is used to generate the UID Resource Table. Refer to the description below on the required parameters for the macro. Once the new source has been created, refer to member UL23309 in the CA-7 SAMPJCL file for applying the USERMOD.

The following is an example of the CA7RTBL macro statement.

```
CA7RTBL RSRC=CA70255,UID=255
```

Where:

CA7RTBL
This is the UID Resource Table generation macro. It is used to build the UID Resource Table.

RSRC
The resource name to be generated in this entry of the table. The resource name can be a 1- to 8-character name which meets site specific external security resource naming conventions.

Note: Resource names must not conflict with existing panel or command names if the default PANEL resource class is used. See the description of the RCLASS keyword on the SECURITY statement in Chapter 3.

UID
The value which will be associated with the resource name supplied on the RSRC= parameter. The value can be from 0 (zero) to 255.

5.7.1.1 Usage Notes

1. The UID Resource Table name must be a valid PDS member name.
2. The CA7RTBL macro must be coded starting in column 10.
3. Duplicate resource names are not allowed, but duplicate UID values are allowed in the table.
4. The UID Resource Table source must be assembled and link-edited into a load library accessible by CA-7.
5. The last entry in the table must be specified with a resource name of LAST (RSRC=LAST) to indicate the end of the table. The UID= parameter is not necessary on the last statement.
6. The UID Resource Table name must be identified on the CA-7 initialization file SECURITY statement using the UID= parameter. Refer to 3.1, “SECURITY Statement” on page 3-2 for a discussion of the SECURITY statement and its associated parameters.
7. The resource names coded in the UID Resource Table must be defined to external security.

5.7.1.2 UID Resource Table - SASSRTBL Source

```
Example UID Resource Table - SASSRTBL Source

         TITLE 'CA-7 EXTERNAL SECURITY UID/RESOURCE TABLE'              00010008
SASSRTBL START 0                                                        00020000
SASSRTBL CA7RTBL RSRC=CA70000,UID=000                                   00040008
         CA7RTBL RSRC=CA70001,UID=001                                   00050008
         CA7RTBL RSRC=CA70002,UID=002                                   00070008
         CA7RTBL RSRC=CA70003,UID=003                                   00090008
         CA7RTBL RSRC=CA70004,UID=004                                   00101008
         CA7RTBL RSRC=CA70005,UID=005                                   00103008
         CA7RTBL RSRC=CA70006,UID=006                                   00105008
         CA7RTBL RSRC=CA70007,UID=007                                   00107008
         CA7RTBL RSRC=CA70008,UID=008                                   00109008
         CA7RTBL RSRC=CA70009,UID=009                                   00109208
         CA7RTBL RSRC=CA70010,UID=010                                   00109408
                    .                                                   00109608
                    .                                                   00109808
                    .                                                   00110008
         CA7RTBL RSRC=CA70254,UID=254                                   00142808
         CA7RTBL RSRC=CA70255,UID=255                                   00142908
         CA7RTBL RSRC=LAST                                              00143008
         END                                                            00150000
```

5.7.2 CA-7 Logon and UID Resource Validation

When a user signs on to CA-7, the user can optionally supply a UID resource name in the UID RESOURCE field on the CA-7 Logon screen. If no UID resource name is supplied, a check is made to see if one is defined in the user's CA-7 profile record. If present, the profile resource name is used as if it were entered on the Logon screen. The USERID and PASSWORD supplied are first validated through external security and then a lookup is performed to determine if the user is defined in the CA-7 Internal Security module.

If the user is defined in the Internal Security module, any UID resource name passed on the Logon screen is ignored. If the user is not defined in the Internal Security module, the UID Resource Table is searched to find a matching resource entry.

If the resource name is not found in the UID Resource Table, the user is signed on to CA-7 with a UID value of 0 (zero). If a matching resource entry is found, a call is made to external security to validate the user's authority to access the resource.

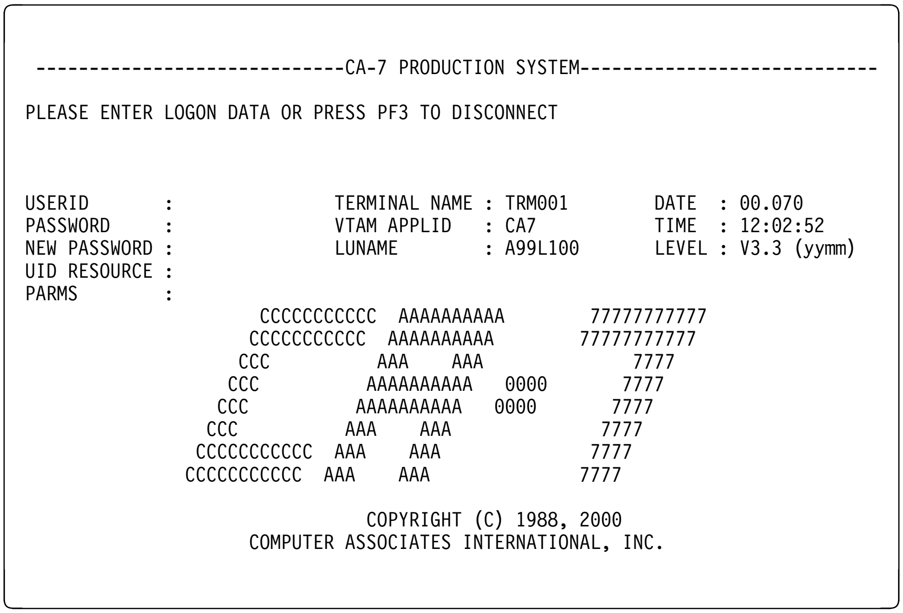
If the user is not authorized to access the resource, a message is displayed indicating the failure and the logon attempt fails. If the user is authorized to access the resource, the associated UID value in the UID Resource Table is assigned to the user.

This process allows external security to control UID assignment to CA-7 users and eliminates the need to maintain all USERIDs in the CA-7 Internal Security module.

Note: The UID resource validation process is the same under the CA-7 ISPF Interface; however, no password is required.

The following is a sample CA-7 Logon screen.

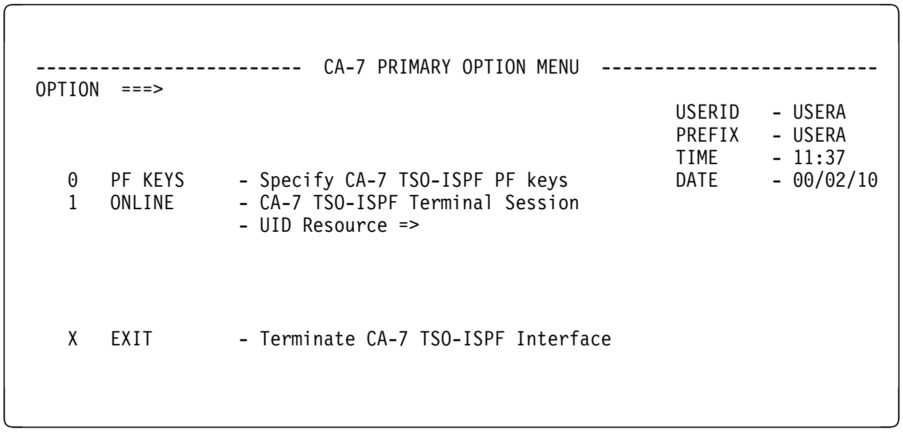
CA-7 Logon Screen



```
 -----------------------------CA-7 PRODUCTION SYSTEM----------------------------

PLEASE ENTER LOGON DATA OR PRESS PF3 TO DISCONNECT



USERID       :               TERMINAL NAME : TRM001        DATE  : 00.070
PASSWORD     :               VTAM APPLID   : CA7           TIME  : 12:02:52
NEW PASSWORD :               LUNAME        : A99L100       LEVEL : V3.3 (yymm)
UID RESOURCE :
PARMS        :
                      CCCCCCCCCCC  AAAAAAAAAA        77777777777
                     CCCCCCCCCCC  AAAAAAAAAA        77777777777
                    CCC          AAA    AAA             7777
                   CCC          AAAAAAAAAA   0000       7777
                  CCC          AAAAAAAAAA   0000       7777
                 CCC          AAA    AAA             7777
                CCCCCCCCCCC  AAA    AAA             7777
               CCCCCCCCCCC  AAA    AAA             7777

                                COPYRIGHT (C) 1988, 2000
                     COMPUTER ASSOCIATES INTERNATIONAL, INC.
```

The following is a sample CA-7 ISPF Interface Primary Option Menu screen.

CA-7 ISPF Interface Primary Option Menu



```
 -------------------------  CA-7 PRIMARY OPTION MENU  --------------------------
 OPTION  ===>
                                                             USERID   - USERA
                                                             PREFIX   - USERA
                                                             TIME     - 11:37
    0   PF KEYS     - Specify CA-7 TSO-ISPF PF keys          DATE     - 00/02/10
    1   ONLINE      - CA-7 TSO-ISPF Terminal Session
                    - UID Resource =>




    X   EXIT        - Terminate CA-7 TSO-ISPF Interface
```

Enter the END command to terminate CA-7 TSO-ISPF.

5.8 /UID Command

The /UID command is used to change a user's current UID processing value through UID resource validation. The /UID command requires that the UID Resource Table option be implemented and that the resources and the appropriate security authorization are defined to external security.

5.8.1 Example

```
      /UID,R=XXXXXXXX
      /UID,LIST
```

Where:

R=
Identifies a resource name which exists in the UID Resource Table. The user's authorization to access the resource is validated through external security and if authorized, the user's current UID value is updated to reflect the UID value found in the UID Resource Table associated with the resource name that was supplied.

LIST
Displays all resource entries and the associated UID values in the UID Resource Table.

5.9 /REFRESH Command

The /REFRESH command is used to refresh the UID Resource Table that was loaded during CA-7 initialization without cycling CA-7. The definitions coded within the specified module will completely replace the definitions currently being used. However, any changes made in the actual CA-ACF2 definitions (using CA-ACF2 commands) may not take effect until CA-7 is recycled.

```
    /REFRESH,MOD=xxxxxxxx
```

Where:

MOD=
Identifies a UID Resource Table in load module format that was built using the CA7RTBL macro.

xxxxxxxx
This must be the member name of the UID Resource Table, and it must reside in a load library accessible to CA-7.

5.10 Program Protection

5.10.1 CA-7 Requirements

CA-7 requires access to numerous programs for execution during production processing. Due to the number of modules involved, a program prefix of SAS can be used when authorizing CA-7 program access. CA-7 also needs access to the main driver module UCC7.

5.10.2 Batch Users

All batch USERIDs which are associated with jobs submitted through CA-7 require access to the program SASSJJCL. This is the LOAD program which identifies resources used by a job to CA-7. To prevent the unauthorized use of CA-7, it is recommended that access to the following programs be secured:

SASSBCLP - Batch Card Load Program
SASSBSTR - Batch Terminal Interface
SASSTRLR - Trailer Step Facility
U7SVC - CA-7 SVC Facility

5.11 External Communicators with CA-ACF2

The external communicators (SASSBSTR, SASSTRLR, U7SVC, and SASSBCLP) provide a means for users outside the CA-7 address space to communicate with CA-7. Because the use of these programs may allow access to production jobs, it is recommended that careful consideration be given to the question of access to these facilities. Users of the Batch Terminal Interface require access to the program SASSBSTR. Users of the Trailer step, the Batch Card Load Program, and U7SVC must be given access to SASSTRLR, SASSBCLP, and U7SVC respectively. Once the question of program access is settled, additional controls may be implemented to prevent unauthorized use of these facilities. This topic describes those controls.

Two types of communication with CA-7 are supported with the external communicators:

- Terminal communication (Batch Terminal Interface, Trailer and U7SVC)
- Data set posting (U7SVC and SASSBCLP)

5.11.1 Terminal Communication

The Batch Terminal Interface (SASSBSTR), the Trailer facility (SASSTRLR), and U7SVC each allow the user to send terminal commands to CA-7. Although no online terminal is used with this mode of communication, input from these programs is treated as terminal input by CA-7. Command security in these environments is handled as it is for all CA-7 terminals. CA-ACF2 will control access to CA-7 commands if EXTERNAL=COMMAND is specified on the SECURITY statement in the CA-7 initialization file. CA-ACF2 will determine a user's access to CA-7 terminal commands based on the LOGONID supplied on the /LOGON command. Thus when using an External Communicator, any command input must be preceded by a CA-7 /LOGON command.

CA-ACF2 normally requires a password at logon. But including passwords in command input for the External Communicators would obviously represent a serious security exposure. Several checks may be made to avoid the need to include passwords in command input when using these facilities. If no /LOGON command is found in the command input, then a /LOGON statement is built using the LOGONID associated with the current user. Under certain conditions it may not be possible to extract the LOGONID associated with the user of the External Communicator. In that event, a /LOGON statement is built using a default LOGONID of CA7DUMMY. If a /LOGON statement is found in the command input then the current user's authority to use the LOGONID found on the /LOGON statement may be checked. If the LOGONID found on the /LOGON statement matches the LOGONID of the current user then it is assumed that the user has the authority to use the LOGONID. If the LOGONIDs differ then a check may be made to validate the user's READ access to a resource whose name is the LOGONID found on the /LOGON statement. The generalized resource type is SUB. Rules for this resource should be written to reflect the security needs of your installation. If a /LOGON statement was generated or if the user's authority to use a LOGONID was successfully validated then CA-7 allows the user to LOGON without a password.

Note: The LOGONID of the current user is determined using CAS9 SSF services. Refer to the Unicenter TNG Framework for OS/390 documentation for more information about SSF.

Submit checking for External Communicators may be activated by modifying ICMDSECT. Refer to the UL233IZ member of SAMPJCL for more information.

5.11.2 SASSTRLR and External Security

The following information is intended to show the actions that are performed by SASSTRLR during execution with relation to security.

A security EXTRACT is done to determine the LOGONID of the submitted job. This LOGONID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied or if a logon statement was supplied without an OPID, then one is generated for the execution. The logon statement resembles the following:

```
  /LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

5.11.3 SASSBSTR and External Security

The following information is intended to show the actions that are performed by SASSBSTR during execution with relation to security.

A security EXTRACT is done to determine the LOGONID of the submitted job. This LOGONID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied, then one is generated for the execution. The logon statement resembles the following:

```
/LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

5.11.4 U7SVC and External Security

The following information is intended to show the actions that are performed by U7SVC during execution with relation to security. There are two different paths that can be taken. It depends on whether there is an input stream with the U7SVC or if it is just a D= to post a data set.

5.11.4.1 U7SVC with D= PARM

A security EXTRACT is done to determine the LOGONID that invoked U7SVC. This LOGONID is later used to generate a full logon statement or optionally perform a data set create check.

If the SVDSNCHK bit is not turned on, the D= command is passed through to CA-7 with no further security checking to be done by U7SVC.

If the SVDSNCHK bit is turned on, then a security call is made by U7SVC. This call determines if the EXTRACTed ID has CREATE authorization for the data set specified on the D=. If the EXTRACTed ID does have authorization, the D= command is passed to CA-7 for processing.

5.11.4.2 U7SVC with an Input Stream

A security EXTRACT is done to determine the LOGONID that invoked U7SVC. This LOGONID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied or if a logon statement was supplied without an OPID, then one is generated for the execution. The logon statement resembles the following:

```
/LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has

EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

5.11.5 Data Set Posting

The Batch Card Load Program (SASSBCLP) and U7SVC allow the user to post the creation of a data set to CA-7. Because such posting may satisfy requirements or cause job triggering, the need to secure the use of these facilities is critical. There are two features of these facilities that should be mentioned in this connection: data set access validation and LOGONID propagation.

The LOGONID associated with the user of U7SVC or SASSBCLP is extracted to determine the user's authority to create the data set. Under certain conditions it may not be possible to extract the LOGONID for the user of the External Communicator. In that event, a default LOGONID of CA7DUMMY is used.

If REQ is specified in the SUBUID hierarchy on the SECURITY statement in the CA-7 initialization file, then the LOGONID associated with the data set creation may be propagated to triggered jobs.

For example, suppose that a user whose LOGONID is XXX submits a batch job which uses U7SVC to post the creation of data set A.B to CA-7. Suppose also that the creation of this data set triggers job Z. Further suppose that REQ is in the first position in the SUBUID hierarchy. In such a case, LOGONID XXX could be propagated to job Z when the job is submitted.

Note: Each of the External Communicators attempts to extract the USERID of the current user, SASSBCLP and U7SVC may be made to verify the authority of the user to create that data set whose creation is to be posted to CA-7. Refer to the UL233IZ member of SAMPJCL for information on ICMDSECT modifications to implement this checking.

5.12 Sample Definitions

The member ACF2SAMP in file 9 (CAI.CA7.SAMPJCL) on the installation tape contains sample ACF2 commands which can be used to secure the CA-7 processing environment under CA-ACF2. The definitions are intended as examples only and should be reviewed and modified to meet your installations security requirements. Once tailored to your sites specifications, the definitions may be used as batch input to CA-ACF2. Refer to the *CA-ACF2 MVS Administrators Guide*, for a discussion on executing CA-ACF2 commands in batch.

Chapter 6. Implementing Security with IBM RACF

This chapter describes the steps necessary to implement the CA-7 External Security Interface with IBM RACF. A working knowledge of the security structure for RACF and its associated command syntax is required. All of the example RACF commands shown in this chapter must be executed under RACF.

Note: The security definitions provided in this section are recommendations for establishing your CA-7 security environment. It is the responsibility of each site to determine if these recommendations meet local auditing and security standards.

6.1 Defining CA-7 Security to RACF

To secure the CA-7 processing environment under RACF, the following steps must be taken.

- Modify the RACF Resource Descriptor Table to include the resource classes required by CA-7.
- Modify the RACF Router Table to identify processing options for the new resource classes.
- Modify the Started Procedures Table to add the CA-7 and CA-7 ICOM started task names.
- Add user profiles for CA-7 and CA-7 ICOM to RACF.
- Define the data set security access requirements for CA-7 and CA-7 ICOM.
- Optionally, define CA-7 as an application resource to be protected by RACF.
- Modify the CA-7 initialization file to specify the security functions to be controlled by RACF.
- Define CA-7 command and panel security access for users to RACF.
- Identify USERID requirements for jobs being submitted by CA-7.

6.2 RACF Requirements

6.2.1 System Requirements

Following are the system requirements for RACF:

- RACF must be at version level 1.7 or above.
- To use the CA-7 External Security Interface with RACF, the CA Standard Security Facility (CAISSF) is required. CAISSF is a subcomponent of the CAIRIM services.

6.2.2 Resource Class Descriptor Table - ICHRRCDE

The Resource Class Descriptor Table is used to identify the general resource classes to be protected by RACF. CA-7 currently requires two resource classes for implementation of the CA-7 External Security Interface with RACF. The IBM macro ICHERCDE is used to define the resource classes to the Resource Class Descriptor Table. Refer to the IBM manual, *System Programming Library: RACF*, for details on updating the Class Descriptor Table.

6.2.2.1 ICHERCDE

ICHERCDE, the IBM supplied Class Descriptor macro, located in SYS1.MACLIB, is used to define installation required resource classes under RACF. Once updated, the source is assembled and link edited as module ICHRRCDE and resides in SYS1.LINKLIB. This module may reside in any link listed library, however, ensure that a copy of this module does not exist in a library anywhere above the new module in the link list libraries concatenation. Refer to the IBM manual, *System Programming Library: RACF*, for details on the ICHRRCDE module and the ICHERCDE macro parameters.

The following entries must be added to the table for CA-7.

The resource classes required by CA-7 are PA@EL and SU@MIT. Refer to the following example:

```
PA@EL    ICHERCDE CLASS=PA@EL,                                   +
               ID=xxx,             available resource number     +
               MAXLNTH=8,                                        +
               FIRST=ALPHANUM,                                   +
               OTHER=ANY,                                        +
               POSIT=xx,           bit position in bitstring     +
               OPER=NO,                                          +
               RACLIST=ALLOWED,                                  +
               GENLIST=ALLOWED
SU@MIT   ICHERCDE CLASS=SU@MIT,                                  +
               ID=xxx,             available resource number     +
               MAXLNTH=8,                                        +
               FIRST=ALPHANUM,                                   +
               OTHER=ANY,                                        +
               POSIT=xx,           bit position in bitstring     +
               OPER=NO,                                          +
               RACLIST=ALLOWED,                                  +
               GENLIST=ALLOWED
         ICHERCDE
```

Note: The last entry must be the ICHERCDE macro with no parameters.

6.2.3 RACF Router Table

The RACF Router Table provides a means to associate installation resource class authorization calls with RACF functions. The resource classes added to the Class Descriptor Table must also be added to the RACF Router Table to specify security processing requirements for the resource classes. Refer to the IBM manual, *System Programming Library: RACF*, for details on implementing or modifying the RACF Router Table.

The following entries must be added to the Router Table to implement the CA-7 External Security Interface with RACF. The entries are added using the IBM ICHRFR01 macro. This module must reside in a link listed library.

6.2.3.1 Syntax

```
PA@EL    ICHRFTRB CLASS=PA@EL,ACTION=RACF
SU@MIT   ICHRFTRB CLASS=SU@MIT,ACTION=RACF
```

Where:

PA@EL and SU@MIT
Specifies the label field which identifies the resource class.

ICHRFTRB
Identifies the IBM RACF Router Table macro.

CLASS=PA@EL
Identifies a resource class of PA@EL.

CLASS=SU@MIT
Identifies a resource class of SU@MIT.

ACTION=RACF
Specifies the action to be taken for this resource class.

Note: The @ is required in the resource class name.

6.2.3.2 Activating the New Resource Classes

To activate the resource classes under RACF, issue the following command:

```
SETROPTS CLASSACT(PA@EL,SU@MIT)
```

6.2.4 Started Procedures Table - ICHRIN03

Started tasks have system generated job statements and do not have associated USER, GROUP, or password parameters. RACF requires a user or group ID to specifically authorize access to resources. The Started Procedures Table for RACF allows installations to associate a USERID with a started task which can then be used to specify authorization access.

Refer to the following example:

```
ICHRIN03 CSECT
         TITLE 'ICHRIN03 - STARTED PROCEDURES TABLE'
         EJECT
         DC    XL2'8003'          NEW FORMAT - 03 ENTRIES
*
         DC    CL8'CA7ONL  '      PROCNAME - CA-7 PRODUCTION ID
         DC    CL8'CA7ONL  '      USERID
         DC    CL8'        '      GROUP - NULL
         DC    XL1'00'            NOT PRIVILEGED OR TRUSTED
         DC    XL7'00'            RESERVED
*
         DC    CL8'CA7ICOM '      PROCNAME
         DC    CL8'CA7ICOM '      USERID
         DC    CL8'        '      GROUP
         DC    XL1'00'            NOT PRIVILEGED OR TRUSTED
         DC    XL7'00'            RESERVED
*
         DC    CL8'*       '      PROCNAME
         DC    CL8'=       '      USERID
         DC    CL8'        '      GROUP
         DC    XL1'00'            NOT PRIVILEGED OR TRUSTED
         DC    XL7'00'            RESERVED
*
         END
```

This table must reside in the link pack area (LPA). The last entry shown is a generic entry which RACF uses if the specific started task name is not found in the table. The equal sign states that the started task name is used as the USERID for any entry that does not match an entry in the table. See the IBM manual, *System Programming Library: RACF*, for a complete description of the Started Procedures Table. Security for the CA-7 and CA-7 ICOM started task do not take effect until the USERIDs are defined to RACF.

6.3 Defining the CA-7 Started Task to RACF

The ADDUSER command is used under RACF to define a new user and to associate that user with an existing RACF defined group.

6.3.1 Syntax

```
ADDUSER CA7ONL NAME('CA-7 ONLINE') OWNER(CA7GROUP) PASSWORD(CA7ONL)
```

Where:

ADDUSER
The RACF command used to define a new user to RACF.

CA7ONL
The name chosen in this example for the CA-7 USERID which is associated with the CA-7 started task.

NAME('CA-7 ONLINE')
The RACF parameter used to describe the USERID.

OWNER(CA7GROUP)
The predefined owning group for the CA-7 USERID profile. The group or USERID used in the OWNER parameter must already be defined to RACF.

PASSWORD
The password chosen in this example for the CA-7 USERID. This provides an additional level of security for the CA-7 started task ID. If no password is specified, the password defaults to the owning group name which may be available to unauthorized personnel.

6.4 Defining CA-7 ICOM to RACF

The CA-7 ICOM, Independent Communications Manager, which manages SMF tracking data for CA-7 must also be defined to RACF. The following example can be used to add the CA-7 ICOM USERID to RACF.

6.4.1 Syntax

```
ADDUSER CA7ICOM NAME('CA-7 ICOM') OWNER(CA7GROUP) PASSWORD(CA7ICOM)
```

Where:

ADDUSER
The RACF command used to define a new user to RACF.

CA7ICOM
The name chosen for the CA-7 Independent Communications Manager (ICOM) started task.

NAME
RACF keyword used to describe the user profile being defined.

OWNER
An existing RACF defined USERID or group which owns this user profile.

PASSWORD
The RACF keyword used to specify a password for the USERID being defined.

> **CA7ICOM**
> The password chosen for the CA7ICOM USERID. If a password is not defined, the owning group name becomes the password by default.

6.5 CA-7 and CA-7 ICOM Data Set Access Requirements

CA-7 requires access to all of the CA-7 permanent data sets defined during installation. CA-7 must also have full access to all JCL libraries which are defined in the CA-7 initialization file. See the *CA-7 Systems Programmer Guide* for a complete description of the CA-7 permanent data sets and the definition of JCL libraries.

6.6 Defining the CA-7 Application Resource Profile

The CA-7 External Security Interface allows you to define CA-7 as an application resource to RACF. The application resource name can then be specified on the SECURITY statement in the CA-7 initialization file. During the LOGON validation process, an additional check is made to determine if the user has the authority to access the CA-7 application resource. This feature is optional; however, it allows for an additional level of security protection for CA-7.

6.6.1 Syntax

```
RDEFINE APPL CA7PROD DATA('CA-7 Security Application Resource')
OWNER(CA7USERS) UACC(NONE)
```

Where:

RDEFINE
The command used to define resources to RACF.

APPL
The resource class name for application resources under RACF.

CA7PROD
The name chosen in this example for the CA-7 security application resource name.

DATA
Used to describe the application resource entry.

OWNER
An existing RACF defined group which owns the resource.

UACC
RACF keyword used to define the universal access for this resource. In this case, NONE.

After defining the CA-7 security application resource to RACF, users must be authorized to access the CA-7 APPL resource. This can be accomplished by using the RACF PERMIT command.

6.6.2 Syntax

```
PERMIT CA7PROD CLASS(APPL) ID(xxxxxxx)
```

Where:

PERMIT
The RACF command used to grant access to resources.

CA7PROD
The name chosen in the previous example for the CA-7 security application resource name.

CLASS(APPL)
The resource class for which this command applies. (Application.)

ID(xxxxxxx)
The USERID you wish to grant access to the CA-7 security application resource.

6.7 Defining CA-7 Command and Panel Security to RACF

Security for CA-7 commands and panels can be protected under RACF by defining each panel and command as a resource. In addition to restricting access to top line commands and panels, functions found on each panel can be protected by specifying an access level for each panel. Refer to Appendix A, “CA-7 Security Tables” on page A-1 for a list of the CA-7 commands, panels, and a cross reference of panel functions with their associated access level requirements.

The following examples illustrate the use of the RACF RDEFINE and PERMIT commands to first define the CA-7 command or panel as a resource to RACF and then "permit" access to specific commands. The resource class is PA@EL for both CA-7 commands and panels.

6.7.1 Syntax

```
RDEFINE PA@EL (L2DB1) DATA('CA-7 Job Definition Panel') OWNER(CA7USERS)
UACC(NONE)
```

Where:

RDEFINE
The RACF command used to define general resources.

PA@EL
The resource class type for CA-7 commands and panels.

(L2DB1)
The resource name for the CA-7 Job Definition panel.

OWNER(CA7USERS)
A predefined RACF user or group profile which owns this resource.

UACC(NONE)
The universal access level for this resource. In this case, NONE.

This example grants access to the resource L2DB1 defined to RACF in the previous example.

6.7.2 Syntax

```
PERMIT L2DB1 CLASS(PA@EL) ID(xxxxxxx) ACCESS(READ,UPDATE)
```

Where:

PERMIT
The RACF command used to grant access to a resource.

L2DB1
The resource name for the CA-7 Job Definition panel.

CLASS(PA@EL)
The resource class type.

ID(xxxxxxx)
The USERID being granted access to the resource.

ACCESS(READ,UPDATE)
The access level for functions found on the Job Definition panel. The user would have full access to functions which require READ and UPDATE authority.

6.8 Securing the /MVS Command

The /MVS command allows a CA-7 user to issue an MVS console command from a CA-7 terminal. Although such a facility may prove indispensable in certain situations, the risks associated with an indiscriminate use of the command are obvious. This section discusses security considerations regarding the use of the command. For additional information about the /MVS command, refer to the *CA-7 Commands Guide*.

The /MVS command text is sent to MVS using SVC 34. The USERID that is associated with the CA-7 address space is the USERID in control when the SVC is issued.

CA-7 does not perform any special validation to verify the terminal user's authority for the MVS command attempted. If the user is allowed to issue the /MVS command, then the specified command text will be sent to MVS.

It is recommended that CA-7 command security be employed to restrict /MVS command access to a limited class of privileged users.

6.9 Controlling Job Submission Under RACF

Depending on your security options, submit checking may be done by CA-7 to validate the authorization of a USERID to submit jobs for another USERID. The 3.1, “SECURITY Statement” on page 3-2 describes the options available to perform submit checking.

This example defines a submit (SU@MIT) resource that can be validated by CA-7.

6.9.1 Syntax

```
RDEFINE SU@MIT (USERID1) DATA('userid1 submission class') OWNER(CA7USERS)
UACC(NONE)
```

Where:

RDEFINE
The RACF command used to define general resources.

SU@MIT
The resource class type for CA-7 submission checking.

(USERID1)
The USERID that can be submitted by other user IDs.

DATA
Used to describe the submission resource class.

OWNER(CA7USERS)
A predefined RACF user or group profile which owns this resource.

UACC(NONE)
The universal access level for this resource. In this case, NONE.

This example grants submit authority for USERID2 to submit for USERID1.

6.9.2 Syntax

```
PERMIT USERID1 CLASS(SU@MIT) ID(USERID2)
```

Where:

PERMIT
The RACF command used to grant access to a resource.

USERID1
The USERID that can be submitted by the ID USERID2 in this example.

CLASS(SU@MIT)
The resource class type.

ID(USERID2)
The USERID that is given submit authority for another ID.

6.10 Surrogate Usage for Job Submission Under RACF

Beginning with RACF 1.9, a surrogate designation can be assigned to USERIDs. This designation allows one USERID to submit jobs on behalf of another USERID. If CA-7 will be submitting jobs that have USERIDs in the JCL that are different from the CA-7 USERID, then a surrogate designation may be needed. This designation would allow CA-7 to submit those jobs with a USERID that is not the same as the one CA-7 uses.

This allows CA-7 to submit jobs with various USERIDs, but this is different from the "Submit Checking" that is done by CA-7. CA-7 can check for submit authority, but it is done using the SU@MIT class not the SURROGATe class.

For more information regarding SURROGATe classes, refer to the *RACF 1.9 Security Administrators Guide*.

This example grants surrogate authority for CA-7 to submit jobs for USERID1.

6.10.1 Syntax

```
PERMIT CLASS(SURROGAT) USERID1.SUBMIT ID(CA7ONL) ACCESS(READ)
```

Where:

PERMIT
The RACF command used to grant access to a resource.

CLASS(SURROGAT)
The resource class type.

USERID1.SUBMIT
The USERID that can be submitted by the ID CA7ONL in this example.

ID(CA7ONL)
The USERID that is given submit authority for another ID.

ACCESS(READ)
The ACCESS level to be given.

6.11 Job Submission

CA-7 maintains a record of USERIDs which can be associated with a job from entry into the CA-7 queues through job submission. CA-7 allows you to specify a hierarchy of candidate USERID sources in the CA-7 initialization file, one of which is propagated to a job during submission, to meet batch security requirements on your system. The candidate USERID sources are:

Job Owner
A USERID specified in the OWNER field for a job on the DB.1 Job Definition panel. (SUBUID value of OWNER)

JCL ID This is a USERID that exists in a job's JCL at entry into the request queue. If a job's JCL contains a USERID at queue entry, USERID insertion does **not** take place. The JCL ID overrides all other USERIDs.

Requester This is the USERID of the user which requests a job through the DEMAND, LOAD, or RUN commands. (SUBUID value of REQ)

Queue JCL
This is the USERID of a user editing a job's queued JCL in the CA-7 request queue. (SUBUID value of QJCL)

CA-7 This is the USERID assigned to CA-7 at startup. If requested the CA-7 ID is propagated to submitted jobs. (SUBUID value of CA7)

The priority of USERID sources is determined by the specification of the SUBUID keyword on the SECURITY statement in the CA-7 initialization file. The SUBUID keyword specifies a hierarchy of USERIDs for JCL insertion.

At submission time, CA-7 scans the USERID hierarchy to determine if a USERID is available from the first hierarchy entry. If an ID is found, it is inserted into the job's JCL and the job is submitted, assuming all other requirements have been met. If an ID was not found, the next source entry is checked for an available ID.

This process continues until an ID is found and inserted into the JCL. If all potential sources have been checked and a USERID is not available, CA-7 checks the status of the SUBNOID flag. The SUBNOID flag is set by the SUBNOID keyword specified on the SECURITY statement in the CA-7 initialization file. If SUBNOID=YES, a job may be submitted without a USERID. If SUBNOID=NO, jobs may not be submitted without a valid USERID and are moved back to the request queue with a requirement status of R-NOUID. The R-NOUID status indicates that all USERID sources were checked and no valid USERID was found for JCL insertion.

6.11.1 RACF USERID Format

For USERID insertion, CA-7 modifies the last statement of the JCL job statement to add the USERID. A comma is added to the last statement to indicate continuation and is followed by a USER= statement to supply the USERID. The following example illustrates the JCL statement format used during ID insertion.

```
// USER=userid
```

Note: The USERID password is inserted, if required, by RACF. This is an automatic function related to Job Submission.

The SUBNOID parameter is used to specify whether a job may be submitted without a USERID. If SUBNOID equals YES, the job is submitted without a USERID. If SUBNOID equals NO, the job is moved back to the request queue with a status of R-NOUID.

The R-NOUID status indicates that all candidate USERID sources in the hierarchy were scanned without finding a USERID for the job and that the SUBNOID parameter requested that jobs not be submitted without a USERID.

6.11.2 Satisfying the R-NOUID Requirement

There are two ways to satisfy the R-NOUID requirement.

- The first method is to edit the job's queue JCL and add a USERID using the JCL USER= parameter. The added USERID is then recognized by CA-7 as an existing JCL USERID and thereby satisfies the R-NOUID requirement.
- The second method requires that the QJCL parameter be specified in the SUBUID hierarchy. Edit the job's JCL and SAVE/REPLACE the JCL without actually making any modifications. Because the QJCL parameter was specified in the SUBUID hierarchy, CA-7 recognizes the queue JCL editor as a potential USERID source and retains the USERID of the user editing the queue JCL. A USERID would now be available and therefore satisfy the R-NOUID requirement.

Note: If a job's JCL contains a USERID at queue entry time, this USERID overrides all other USERIDs and USERID insertion will not take place.

You may not manually satisfy or POST an R-NOUID requirement. If you try to satisfy the requirement by any method other than those listed previously, the request is ignored.

6.11.3 USERID Propagation

Establishing USERIDs to be associated with jobs when submitted by CA-7 is a critical aspect of security. Defining the USERID hierarchy requires careful planning to ensure that each job is submitted with the correct ID and therefore the proper security. If Requester (REQ) is specified in the SUBUID hierarchy, USERIDs are propagated to any jobs triggered by the original request. This means that USERID propagation of a requesting ID occurs for the following conditions:

- The USERID of a user requesting work through the DEMAND, LOAD or RUN commands.
- The USERID associated with a job which triggers any additional jobs.
- For data set triggers, the USERID associated with a job which was submitted by CA-7 and created the data set.
- For data set triggers which are initiated through U7SVC and SASSBCLP, the USERID associated with the user posting the creation of the data set.

6.12 UID Resources

Access to information on the CA-7 database is controlled through UID security validation. When a user attempts to access a job on the database, regardless of whether internal or external security is in control, the user's UID value is checked against the UID value associated with the job. This provides job-level security for the CA-7 database. External security does not provide an equivalent JOB level protection; therefore, it is important that each CA-7 user be assigned a UID which relates to the user's area of responsibility.

Note: UID resource security is only valid in an environment where external security controls CA-7 logons. Calls are made to the external security package to validate a user's authority to access the resource. The resources have no meaning to CA-7 internal security.

The UID value (0-255) can be obtained from the USERID entry in the CA-7 internal security module, through UID resource validation during logons to CA-7 or through the /UID top line command issued under a CA-7 session. If you wish to maintain USERIDs in the CA-7 internal security module, refer to Chapter 7, “CA-7 Internal Security” on page 7-1 for details on defining USERIDs in the internal security module. The following information outlines the steps necessary to implement UID resource validation and describes the security processing involved.

UID resource security requires a UID Resource Table which CA-7 references during UID resource validation. This table contains resource names and associated UID value entries to be used during the UID validation process. A sample resource table, SASSRTBL, is provided in both source and load module format and may be used to implement UID resource security. To create a site specific UID Resource Table with unique resource names and UID values, use the CA7RTBL macro to generate the table.

Note: The default resource class for UID resources is PA@EL. You may change the resource class for calls to external security using the RCLASS= parameter on the CA-7 initialization file SECURITY statement.

You may use the /PROF command to establish and maintain a default UID resource for users logging on to CA-7. Refer to the *CA-7 Commands Guide* for a description of the /PROF command.

6.12.1 CA7RTBL Macro

The CA7RTBL macro is used to generate the UID Resource Table. Refer to the description below on the required parameters for the macro. Once the new source has been created, refer to member UL23309 in the CA-7 SAMPJCL file for applying the USERMOD.

The following is an example of the CA7RTBL macro statement.

```
CA7RTBL RSRC=CA70255,UID=255
```

Where:

CA7RTBL
This is the UID Resource Table generation macro. It is used to build the UID Resource Table.

RSRC
The resource name to be generated in this entry of the table. The resource name can be a 1- to 8-character name which meets site specific external security resource naming conventions.

Note: Resource names must not conflict with existing panel or command names if the default PANEL resource class is used. See the description of the RCLASS keyword on the SECURITY statement in Chapter 3.

UID
The value which will be associated with the resource name supplied on the RSRC= parameter. The value can be from 0 (zero) to 255.

6.12.1.1 Usage Notes

1. The UID Resource Table name must be a valid PDS member name.
2. The CA7RTBL macro must be coded starting in column 10.
3. Duplicate resource names are not allowed, but duplicate UID values are allowed in the table.
4. The UID Resource Table source must be assembled and link-edited into a load library accessible by CA-7.
5. The last entry in the table must be specified with a resource name of LAST (RSRC=LAST) to indicate the end of the table. The UID= parameter is not necessary on the last statement.
6. The UID Resource Table name must be identified on the CA-7 initialization file SECURITY statement using the UID= parameter. Refer to 3.1, "SECURITY Statement" on page 3-2 for a discussion of the SECURITY statement and its associated parameters.
7. The resource names coded in the UID Resource Table must be defined to external security.

6.12.1.2 UID Resource Table - SASSRTBL Source

```
Example UID Resource Table - SASSRTBL Source

         TITLE 'CA-7 EXTERNAL SECURITY UID/RESOURCE TABLE'              00010008
SASSRTBL START 0                                                        00020000
SASSRTBL CA7RTBL RSRC=CA70000,UID=000                                   00040008
         CA7RTBL RSRC=CA70001,UID=001                                   00050008
         CA7RTBL RSRC=CA70002,UID=002                                   00070008
         CA7RTBL RSRC=CA70003,UID=003                                   00090008
         CA7RTBL RSRC=CA70004,UID=004                                   00101008
         CA7RTBL RSRC=CA70005,UID=005                                   00103008
         CA7RTBL RSRC=CA70006,UID=006                                   00105008
         CA7RTBL RSRC=CA70007,UID=007                                   00107008
         CA7RTBL RSRC=CA70008,UID=008                                   00109008
         CA7RTBL RSRC=CA70009,UID=009                                   00109208
         CA7RTBL RSRC=CA70010,UID=010                                   00109408
                   .                                                    00109608
                   .                                                    00109808
                   .                                                    00110008
         CA7RTBL RSRC=CA70254,UID=254                                   00142808
         CA7RTBL RSRC=CA70255,UID=255                                   00142908
         CA7RTBL RSRC=LAST                                              00143008
         END                                                            00150000
```

6.12.2 CA-7 Logon and UID Resource Validation

When a user signs on to CA-7, the user can optionally supply a UID resource name in the UID RESOURCE field on the CA-7 Logon screen. If no UID resource name is supplied, a check is made to see if one is defined in the user's CA-7 profile record. If present, the profile resource name is used as if it were entered on the Logon screen. The USERID and PASSWORD supplied are first validated through external security and then a lookup is performed to determine if the user is defined in the CA-7 Internal Security module.

If the user is defined in the Internal Security module, any UID resource name passed on the Logon screen is ignored. If the user is not defined in the Internal Security module, the UID Resource Table is searched to find a matching resource entry.

If the resource name is not found in the UID Resource Table, the user is signed on to CA-7 with a UID value of 0 (zero). If a matching resource entry is found, a call is made to external security to validate the user's authority to access the resource.

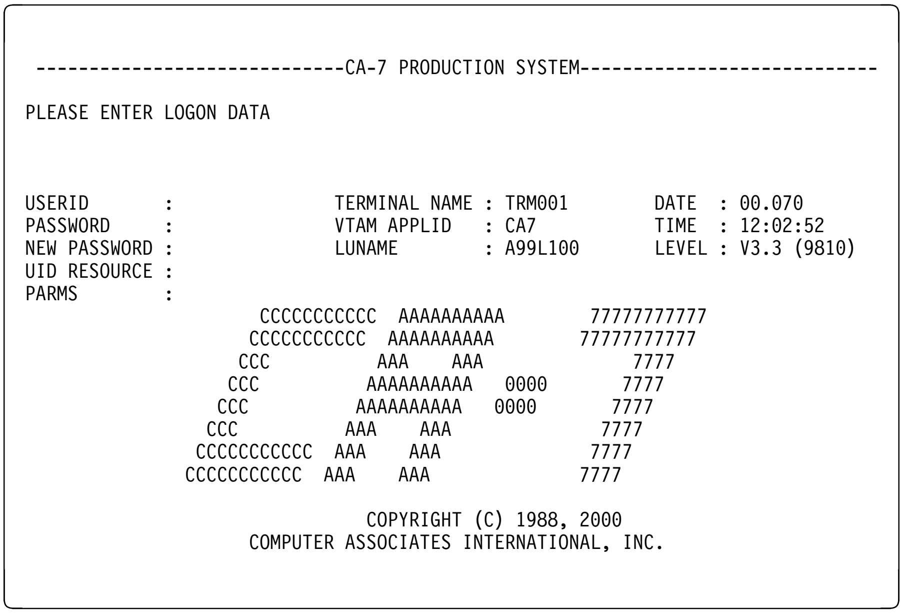
If the user is not authorized to access the resource, a message is displayed indicating the failure and the logon attempt fails. If the user is authorized to access the resource, the associated UID value in the UID Resource Table is assigned to the user.

This process allows external security to control UID assignment to CA-7 users and eliminates the need to maintain all USERIDs in the CA-7 Internal Security module.

Note: The UID resource validation process is the same under the CA-7 ISPF Interface; however, no password is required.

The following is a sample CA-7 Logon screen.

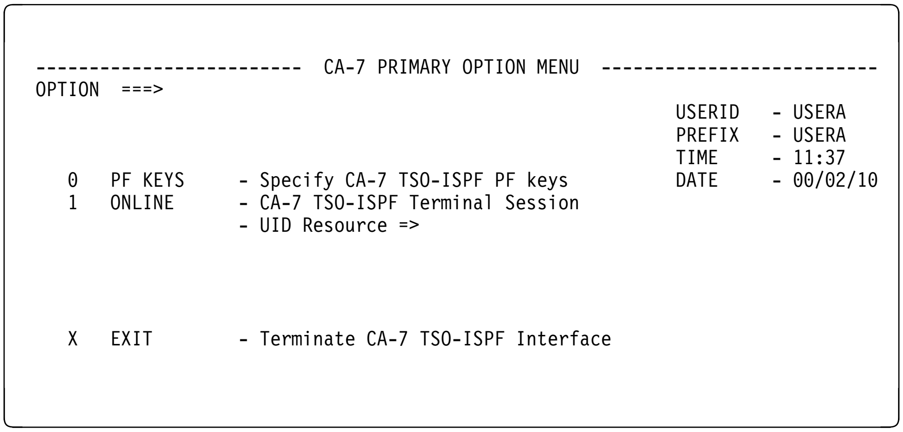
CA-7 Logon Screen



```
 -----------------------------CA-7 PRODUCTION SYSTEM----------------------------

PLEASE ENTER LOGON DATA



USERID       :               TERMINAL NAME : TRM001        DATE  : 00.070
PASSWORD     :               VTAM APPLID   : CA7           TIME  : 12:02:52
NEW PASSWORD :               LUNAME        : A99L100       LEVEL : V3.3 (9810)
UID RESOURCE :
PARMS        :
                      CCCCCCCCCCC  AAAAAAAAAA        77777777777
                     CCCCCCCCCCC  AAAAAAAAAA        77777777777
                    CCC          AAA    AAA             7777
                   CCC          AAAAAAAAAA   0000       7777
                  CCC          AAAAAAAAAA   0000       7777
                 CCC          AAA    AAA             7777
                CCCCCCCCCCC  AAA    AAA             7777
               CCCCCCCCCCC  AAA    AAA             7777

                                COPYRIGHT (C) 1988, 2000
                     COMPUTER ASSOCIATES INTERNATIONAL, INC.
```

The following is a sample CA-7 ISPF Interface Primary Option Menu screen.

CA-7 ISPF Interface Primary Option Menu



```
 -------------------------  CA-7 PRIMARY OPTION MENU  --------------------------
 OPTION  ===>
                                                            USERID   - USERA
                                                            PREFIX   - USERA
                                                            TIME     - 11:37
    0   PF KEYS     - Specify CA-7 TSO-ISPF PF keys         DATE     - 00/02/10
    1   ONLINE      - CA-7 TSO-ISPF Terminal Session
                    - UID Resource =>




    X   EXIT        - Terminate CA-7 TSO-ISPF Interface
```

Enter the END command to terminate CA-7 TSO-ISPF.

6.13 /UID Command

The /UID command is used to change a user's current UID processing value through UID resource validation. The /UID command requires that the UID Resource Table option be implemented and that the resources and the appropriate security authorization are defined to external security.

6.13.1 Syntax

```
/UID
►►──/UID──,──┬─R=resname─┬──────────────────────────────────────►◄
             └─LIST──────┘
```

Where:

R=

Identifies a resource name which exists in the UID Resource Table. The user's authorization to access the resource is validated through external security and if authorized, the user's current UID value is updated to reflect the UID value found in the UID Resource Table associated with the resource name that was supplied.

LIST

Displays all resource entries and the associated UID values in the UID Resource Table.

6.14 /REFRESH Command

The /REFRESH command is used to refresh the UID Resource Table that was loaded during CA-7 initialization without cycling CA-7. The definitions coded within the specified module will completely replace the definitions currently being used. However, any changes made in the actual RACF definitions (using RACF commands) may not take effect until CA-7 is recycled.

6.14.1 Syntax

/REFRESH

```
►►──/REFRESH──,──MOD──=──xxxxxxxx─────────────────────────────────────►◄
```

Where:

MOD=

Identifies a UID Resource Table in load module format that was built using the CA7RTBL macro.

xxxxxxxx

This must be the member name of the UID Resource Table, and it must reside in a load library accessible to CA-7.

6.15 Program Protection

6.15.1 CA-7 Requirements

CA-7 requires access to numerous programs for execution during production processing. Due to the number of modules involved, a program prefix of SAS can be used when authorizing CA-7 program access. CA-7 also needs access to the main driver module UCC7.

6.15.2 Batch Users

All batch USERIDs which are associated with jobs submitted through CA-7 require access to the program SASSJJCL. This is the LOAD program which identifies resources used by a job to CA-7. To prevent the unauthorized use of CA-7, it is recommended that access to the following programs be secured:

SASSBCLP - Batch Card Load Program
SASSBSTR - Batch Terminal Interface
SASSTRLR - Trailer Step Facility
U7SVC - CA-7 SVC Facility

6.16 Group Profiles

RACF allows you to define Group Profiles which are used to organize security access at a group level rather than by individual users. Although CA-7 does not support group identification at logon, you can structure your resource authorizations by group and then "connect" users to the specific Group Profiles. If you use multiple "connect groups" you must activate the List-of-Groups authorization checking feature in RACF. Refer to the *RACF Security Administrators Guide* for more details on Group level access.

6.17 External Communicators with IBM-RACF

The External Communicators (SASSBSTR, SASSTRLR, U7SVC, and SASSBCLP) provide a means for users outside the CA-7 address space to communicate with CA-7. Because the use of these programs may allow access to production jobs, it is recommended that careful consideration be given to the question of access to these facilities. Users of the Batch Terminal Interface require access to the program SASSBSTR. Users of the Trailer step, the Batch Card Load Program, and U7SVC must be given access to SASSTRLR, SASSBCLP and U7SVC respectively. Once the question of program access is settled, additional controls may be implemented to prevent unauthorized use of these facilities. This section describes those controls.

Two types of communication with CA-7 are supported with the External Communicators:

- Terminal communication (Batch Terminal Interface, Trailer and U7SVC)
- Data set posting (U7SVC and SASSBCLP)

6.17.1 Terminal Communication

The Batch Terminal Interface (SASSBSTR), the Trailer facility (SASSTRLR), and U7SVC each allow the user to send terminal commands to CA-7. Although no online terminal is used with this mode of communication, input from these programs is treated as terminal input by CA-7. Command security in these environments is handled as it is for all CA-7 terminals. IBM-RACF controls access to CA-7 commands if EXTERNAL=COMMAND is specified on the SECURITY statement in the CA-7 initialization file. IBM-RACF determines a user's access to CA-7 terminal commands based on the USERID supplied on the /LOGON command. Thus when using an External Communicator, any command input must be preceded by a /LOGON command.

IBM-RACF normally requires a password at logon. But including passwords in command input for the External Communicators would obviously represent a serious security exposure. Several checks may be made to avoid the need to include passwords in command input when using these facilities. If no /LOGON command is found in the command input, then a /LOGON statement is built using the USERID associated with the current user. Under certain conditions it may not be possible to extract the USERID associated with the user of the External Communicator. In that event, a /LOGON statement is built using a default USERID of CA7DUMMY. If a /LOGON statement is found in the command input then the current user's authority to use the USERID found on the /LOGON statement may be checked. If the USERID found on the /LOGON statement matches the USERID of the current user then it is assumed that the user has the authority to use the USERID. If the USERIDs differ then a check may be made to validate the user's READ access to an entity whose name is the USERID found on the /LOGON statement. The resource class is SU@MIT. Security definitions for this resource should be created to reflect the security needs of your installation. If a /LOGON statement was generated or if the user's authority to use a USERID was successfully validated then CA-7 allows the user to LOGON without a password.

Note: The USERID of the current user will be determined using CAS9 SSF services. Refer to the Unicenter TNG Framework for OS/390 documentation for more information about SSF.

Submit checking for External Communicators may be activated by modifying ICMDSECT. Refer to the UL233IZ member of SAMPJCL for more information.

6.17.2 SASSTRLR and External Security

The following information is intended to show the actions that are performed by SASSTRLR during execution with relation to security.

A security EXTRACT is done to determine the user ID of the submitted job. This user ID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied or if a logon statement was supplied without an OPID, then one is generated for the execution. The logon statement resembles the following:

```
/LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

6.17.3 SASSBSTR and External Security

The following information is intended to show the actions that are performed by SASSBSTR during execution with relation to security.

A security EXTRACT is done to determine the user ID of the submitted job. This user ID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied, then one is generated for the execution. The logon statement resembles the following:

```
    /LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

6.17.4 U7SVC and External Security

The following information is intended to show the actions that are performed by U7SVC during execution with relation to security. There are two different paths that can be taken. It depends on whether there is an input stream with the U7SVC or if it is just a D= to post a data set.

6.17.4.1 U7SVC with D= PARM

A security EXTRACT is done to determine the user ID that invoked U7SVC. This user ID is later used to generate a full logon statement or optionally perform a data set create check.

If the SVDSNCHK bit is not turned on, the D= command is passed through to CA-7 with no further security checking to be done by U7SVC.

If the SVDSNCHK bit is turned on, then a security call is made by U7SVC. This call determines if the EXTRACTed ID has CREATE authorization for the data set specified on the D=. If the EXTRACTed ID does have authorization, the D= command is passed to CA-7 for processing.

6.17.4.2 U7SVC with an Input Stream

A security EXTRACT is done to determine the user ID that invoked U7SVC. This user ID is later used to generate a full logon statement or optionally perform a submit check. The input stream is then read to determine if a logon statement was supplied. If no logon statement was supplied or if a logon statement was supplied without an OPID, then one is generated for the execution. The logon statement resembles the following:

```
/LOGON extid                                    *GENERATED LOGON*
```

The extid is the EXTRACTed ID of the job. This logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If a logon statement with an OPID is found and the OPID is the same as the EXTRACTed ID, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the OPID is not the same as the EXTRACTed ID, other checks are done. A check is made to see if the BSUBCHK bit is turned on in ICMDSECT. If it is on, then a submit check is performed. The check is done to see if the EXTRACTed ID has the authority to submit on behalf of the OPID in the logon statement. If the check is okay, the logon statement is passed to CA-7 indicating that no password is needed with this particular logon attempt.

If the BSUBCHK bit is turned off, then no submit checks are done. The logon statement is passed as it is coded with no special indication to CA-7. If CA-7 has

EXTERNAL=LOGON coded in the initialization file, a logon check is performed trying to supply a password. If the password was entered on the logon statement, it is validated by the external security package. If no password was coded, then the logon fails due to a missing password.

In general, there are only two times a password is needed:

1. User exit SASSXXLX is coded by the user to require a password.
2. The BSUBCHK bit is NOT on and the OPID is different from the EXTRACTed ID.

Note: The SVDSNCHK and the BSUBCHK bits are defined in ICMDSECT at offset +06. This is the ICM3 byte. The status of this byte can be checked by a request to ICOM. To display the first few bytes of the loaded ICMDSECT, issue the D=DSECT command to ICOM.

6.17.5 Data Set Posting

The Batch Card Load Program (SASSBCLP) and U7SVC allow the user to post the creation of a data set to CA-7. Because such posting may satisfy requirements or cause job triggering, the need to secure the use of these facilities is critical. There are two features of these facilities that should be mentioned in this connection: data set access validation and USERID propagation.

The USERID associated with the user of U7SVC or SASSBCLP is extracted to determine the user's authority to create the data set. Under certain conditions it may not be possible to extract the USERID for the user of the External Communicator. In that event, a default USERID of CA7DUMMY is used.

If REQ is specified in the SUBUID hierarchy on the SECURITY statement in the CA-7 initialization file, then the USERID associated with the data set creation may be propagated to triggered jobs.

For example, suppose that a user whose USERID is XXX submits a batch job which uses U7SVC to post the creation of data set A.B to CA-7. Suppose also that the creation of this data set triggers job Z. Further suppose that REQ is in the first position in the SUBUID hierarchy. In such a case, USERID XXX could be propagated to job Z when the job is submitted.

Note: Each of the External Communicators attempts to extract the USERID of the current user, SASSBCLP and U7SVC may be made to verify the authority of the user to create that data set whose creation is to be posted to CA-7. Refer to the UL233IZ member of SAMPJCL for information on ICMDSECT modifications to implement this checking.

6.18 Sample Definitions

The member RACFSAMP in file 9 (CAI.CA7.SAMPJCL) on the installation tape contains sample RACF commands which can be used to secure the CA-7 processing environment under RACF. The definitions are intended as examples only and should be reviewed and modified to meet your installations security requirements. Once tailored to your site's specifications, the definitions may be used as batch input to RACF. Refer to the *RACF Command Language Reference Manual* for a discussion on executing RACF commands in batch.

Chapter 7. CA-7 Internal Security

The user can define the security structure and level of authority for each individual allowed access to CA-7. You should make a careful evaluation of security requirements prior to implementation of CA-7.

The ability to control levels of authority is based on the CA-7 SECURITY macro. You can define five distinct levels of security:

- Terminal/Operator
- Operator/CA-7 Application
- CA-7 Application/Command
- Command/Function
- User ID/External Data Set

Each level of security or authorization provides further qualification (or restriction) of the preceding level. This means each level of security defined by the SECURITY macro requires that preceding levels also be defined. For example, to have authorization to perform a specific command within an application (Application/Command security), an operator must also have authorization to use the application providing that command. You must define the first level of security (Terminal/Operator) before an operator is allowed to log on to CA-7.

Establishing the user-defined security structure is accomplished through the CA-7 initialization process. The CA-7 initialization file must contain a SECURITY statement. This SECURITY statement points to a load module containing the user's security definitions. The Security module identifies all operators authorized to log on to CA-7, which terminal each operator can use, and other authorization qualifiers necessary to limit the interface with CA-7. The definition of control provided can be modified whenever necessary because of changes in personnel, addition/deletion of terminals, and so forth. Since the initialization file pointing to this matrix is reloaded each time CA-7 is initialized, the most current definition of the data center's security structure is always in effect.

7.1 Security Use Considerations

Capabilities provided by CA-7 make it important to identify control personnel within the organization who will be allowed to interface directly with CA-7. By establishing a security structure, various access levels can be allowed and still controlled, making CA-7 available for use by any number of authorized personnel. CA-7 security can be used in the following ways:

- The Master Terminal Operator (MTO) has the most responsibility for the activity of CA-7. MTO functions might include monitoring production activity, providing status reports, responding to CA-7's requests for JCL overrides or manual verifications, and using various application functions to ensure database integrity.

  To permit the MTO to perform these various functions, the MTO's SECURITY macro statement would allow access to all functions within all CA-7 applications. Assigning a function level of 15 within each application allows this access. In addition, assigning the MTO a special user ID of 255 allows access to all data regardless of ownership.

- The workstation terminal operator generally requires less authority than the MTO. Each operator would be assigned a level of authorization to reflect the various responsibilities of each workstation. Likewise, an individual may have different levels of authorization depending on which terminal is being used and which functions are being performed.

- Database Maintenance may be established as a separate activity to be controlled and performed by one person or a designated group of personnel. The level of authority required for this activity must obviously allow full use of maintenance application functions, but probably would not require the ability to modify the queues.

- End users and some data center managers may also have a need to access CA-7 in inquire-only mode. For example, RJE users could be authorized to interrogate the queues to determine job status and also have access to database entries for those jobs belonging to them as defined by their user ID (UID) value. Data center managers could be authorized to interrogate CA-7 for information reflecting current production status, projected schedules, past history, and so forth.

Regardless of a person's level of responsibility as defined to CA-7, all authorized operators (MTO through end user) must use the same procedure to establish communication with CA-7. This procedure is the logon and is described in the *CA-7 Commands Guide*.

7.2 Defining Security Levels

Once requirements are defined, two general areas of activity need to be reviewed and planned:

- SECURITY macros are used to generate the load module referenced in the initialization file. The default Security module distributed with CA-7 will not restrict security.
- If the need for joint ownership of classified jobs or external data sets has been identified, UIDs need to be established with an optional USERID Security module.

The following provides a description of each of the five levels of security:

7.2.1 Terminal/Operator Security

To enforce the security structure defined, CA-7 requires identification of all personnel (terminal operators). Each terminal operator is identified by a unique operator ID, up to 8 characters, which is specified in the SECURITY macro. The macro must also specify the terminal ID(s) of each terminal on which this operator is allowed to log on (/LOGON).

7.2.2 Operator/Application Security

An operator can be restricted to the use of only those applications which fall within the operator's specific area of responsibility. The SECURITY macro is used to define access to applications for each operator. Application security levels are shown in the chart on Table 7-1 on page 7-4.

7.2.3 Application/Command Security

Within each CA-7 application there is at least one command which can be performed. Each command is assigned a required authorization value (level) from 0 to 15 in the SASSTRAN module. This program may be modified to adjust these values. Any command may be disabled by assigning it a level greater than 15 in SASSTRAN. Commands assigned lower numbers are less restrictive than commands assigned higher numbers. For example, inquiry commands may have level 4, while Utilities may have level 10. An operator is limited to commands of the application that have an assigned value (in SASSTRAN) equal to or less than the authorization level specified in the SECURITY macro for the operator. An authorization level is specified for each application that an operator can access. See Table 7-1 on page 7-4 for each CA-7 application, its commands and assigned levels.

Table 7-1 (Page 1 of 3). Applications and Command Levels Chart

Application ID	**Function Authority Level**	**Function Name**
AR (Automated Recovery Facility)	00	AR.3
MLR (Management Level Reporting)	00	GRAPHD, GRAPHJ, GRAPHN, GRAPHS
PS (Personal Scheduling)	00	PS (See SASSDSCR source module for function security levels)
RSC (Virtual Resource Management)	00	RM (See SASSDSCR source module for function security levels)
SAN (Analyze)	05	PRRNJCL, RESANL, RQMT, RQVER, TRIG, XREF
SCM (System Commands)	00	/CLOSE, /COPY, /DISPLAY, /ECHO, /FETCH, /NXTMSG, /PAGE, /PAnn, /PFnn
	01	/OPERID
	05	/LOG, /MSG, /AUTO
	10	/BRO, /PURGPG, /WTO
	12	/START, /STOP, /SWAP
	13	/RESET
	14	/ASSIGN, /CHANGE, /CLOSE(T=), /DMP1, /DUMP, /LOGOFF(T=), /MVS, /OPEN, /RELINK, /SHUTDOWN, /WLB
	15	/JCL, /OPERIDS, /PROCS
SDM (Database Maintenance)	00	DBM, DSN, PROSE, PROSE(DD), PROSE(DSN), PROSE(JOB), PROSE(NETWORK), PROSE(JOB), PROSE(USER), JCL, JOB, JOBCONN, JOBCONN(DSN), JOBCONN(JDEP), JOBCONN(NWK), JOBCONN(RPT), JOBCONN(USR), NETWORK, QJCL, SCHD(DTRG), SCHD(INWK), SCHD(JOB), SCHD(JTRG), SCHD(NTRG), SCHD(ONWK), SCHDMOD, (See SASSDSCR source module for function security levels)
SFC (Forecast)	04	FALL, FJOB, FPOST, FPRE, FQALL, FQJOB, FQPOST, FQPRE, FQRES, FQSTN, FQTAPE, FRES, FRJOB, FRQJOB, FSTN, FSTRUC, FTAPE, FWLP

Table 7-1 (Page 2 of 3). Applications and Command Levels Chart

Application ID	Function Authority Level	Function Name
SJR (Job Restart)	10	LIST, RESTART
SLI (Inquiry and Report)	00	HELP
	01	LACT(R), LARF, LARFQ, LCTLG, LDSN, LDTM, LJES, LJOB(R), LLOCK, LNTWK, LOC, LPOST, LPRE, LPRRN, LPROS, LQ(P/R), LRDY(P/R), LREQ(P/R), LRES, LRLOG, LRMD, LSCHD, LSYS, LWLB
	04	LJCK, LJCL, LLIB, LPDS
	09	DUMP
SPO (Queue Posting)	05	IN, IO, LOGIN, LOGOUT, OUT, REMIND, RSVP
	10	ADDRQ, ADDSCH, ARFP, CANCEL, CTLG, DEMAND(H), DIRECT, DMDNW, HOLD, JCLOVRD, LOAD(H), NOPRMP, NXTCYC, POST, PRMP, PRSCF, PRSQA, PRSQD, RELEASE, REQUEUE, RESCHNG, RUN(H), RUNNW, RUSH, SUBMIT, SUBSCH, SUBTM, VERIFY
	15	SSCAN, START, STOP
SQM (Queue Maintenance)	00	XPOST, XPRE, XQ, XQJ, XQN, XRQ, XRST, XSPOST, XSPRE, XUPD, XWLB, (See SASSDSCR source module for function security levels)
SRC (Schedule Resolution)	02	PRINT
	04	RESOLV
UTL (Utilities)	00	DMPCAT, DMPDSCB, DMPDSN, FIND, LISTDIR, MAP, SPACE
	04	TIQ
	08	ARTS
	10	AL, ALC, ALLOC, BLDG, CAT, CONN, DEALLOC, DCONN, DLTX, RENAME, SCRATCH, UNC
	15	SCRATCHP, TIQU
SYS (System Information)	00	SYSDMP, SYSINQ

Table 7-1 (Page 3 of 3). Applications and Command Levels Chart

Application ID	**Function Authority Level**	**Function Name**
SCO (Core Manipulation)	00	DMP, ZAP
TRA (System Debugging)	00	DM, FIX, FRE, GO, LTR, PAT, SAV, TRP, ZA (Used only by CA-7 Technical Support representatives.)

7.2.4 Command/Function Screen Security

In a broad application area, such as Database Maintenance (SDM0), a single authorization level as defined in the SASSTRAN module is not enough to control database access and update. Therefore, a security method was devised to control access based on screen, function and terminal.

The security table is defined in SASSDSCR. It is composed of the SECURITY macros.

7.2.4.1 Syntax

```
┌─ SEC ──────────────────────────────────────────────────────────────────────────┐
│ ►►──SEC SCR=nnn──,PAN=nnnnnnnn──┬─────────────────────┬─────────────────►      │
│                                 │          ┌─0──┐     │                        │
│                                 └─,READ=───┴─nn─┴─────┘                        │
│                                                                                │
│ ►──┬──────────────────┬──┬──────────────────┬──┬──────────────────┬─────►      │
│    │       ┌─0──┐     │  │       ┌─0──┐     │  │       ┌─0──┐     │            │
│    └─,ADD=─┴─nn─┴─────┘  └─,UPD=─┴─nn─┴─────┘  └─,DEL=─┴─nn─┴─────┘            │
│                                                                                │
│ ►──┬──────────────────┬──┬───────────────┬──┬─────────────────────┬─────►◄     │
│    │        ┌─0──┐    │  └─,TERMLVL=nn───┘  │           ┌─,───┐   │            │
│    └─,SUBM=─┴─nn─┴────┘                     └─,TERM=(───▼termn┴─)─┘            │
└────────────────────────────────────────────────────────────────────────────────┘
```

7.2.4.2 Parameter Descriptions

SCR=nnn
Identifies the screen. Required. This should not be changed.

PAN=nnnnnnnn
Identifies the panel ID. Required. This should not be changed.

READ=nn
Identifies the authorization level (from 0 to 15) which is required for read-only functions such as LIST, FETCH, and so forth. The default value is 0.

ADD=nn
Identifies the authorization level (from 0 to 15) which is required for add type functions such as ADD or SAVE. The default value is 0.

UPD=nn
Identifies the authorization level (from 0 to 15) which is required for update type functions such as UPD or REPL. The default value is 0.

DEL=nn
Identifies the authorization level (from 0 to 15) which is required for delete type functions such as DELETE or DD. The default value is 0.

SUBM=nn
Identifies the authorization level (from 0 to 15) which is required for job submission functions such as RUN or SUBMIT. The default value is 0.

The TERMLVL and TERM values are optional, but may be used to restrict access by physical terminal.

TERMLVL=nn
Identifies the authorization level (from 0 to 15) at which the terminal list will be checked for access qualification. This allows you to restrict certain screen functions such as UPD or DELETE to specific terminals, but allow functions such as LIST to be performed without terminal restrictions.

TERM=(term1,...,termn)
Identifies one or more terminals to be validated when checking function access. The maximum number of characters that can be within the parentheses is 255. If this limit is encountered, then specify as many terminal names as will fit. Then code another SEC macro immediately after this one, with the terminals listed in the TERM parameter. For this secondary SEC macro, only the TERM parameter can be coded.

The following logic is employed to control access:

- If the function does not access the database (such as CLEAR, EDIT or FORMAT), no security checking is done.
- If the user has an authorization level of 15 defined in the Security module, he is allowed to execute all functions.
- In all other cases, the authorization level as defined in the Security module is compared against the security level required for the appropriate screen and function (READ, ADD, and so forth). If the authorization level is less, the command is rejected.
- The TERM list is now checked for terminal restrictions. If there are none, the command is accepted. If the security level required is less than the TERMLVL value, the command is accepted. Otherwise, the TERM list is checked. If the user's terminal is not found in this list, the command is rejected.

If EXTERNAL=COMMAND was specified on the SECURITY statement in the CA-7 initialization file, then SASSDSCR should not be modified. If CA-7 native security controls command access, then SASSDSCR may be modified to reflect the needs of your installation.

7.2.5 UID/External Data Set Security

The UID/External Data Set security option is used to ensure that only authorized personnel have access to specific classified jobs and identified data sets external to CA-7. This restriction is accomplished for jobs by assigning ownership in the form of a UID through the Database Maintenance application when the jobs are placed under control of CA-7. External data sets can receive similar protection by identifying the data set by name and the UID valid for access in a user-defined USERID Security module.

For a terminal operator to gain access to UID protected jobs or data sets, the operator's SECURITY macro definition must specify a UID matching the UID of the job or the data set defined in the USERID Security module. Since read-only or write-only access may be needed for different UIDs and equivalence between some UIDs may be desirable, a USERID Security module can further define these equivalencies and access limitations. The module is identified on the SECURITY statement of the initialization file with the USER operand.

7.3 SECURITY Macro

The SECURITY macro defines the access levels for all personnel who will be designated as CA-7 operators as follows:

- The terminals where the specified ID is allowed access.
- The operator IDs (OPID) used for logon.
- The CA-7 applications the operator will have access to and the function level within each.
- Operator restriction to only those jobs which carry a specific UID or ownership code.

A Security module should be assembled and link edited for the user's specific environment. It must not be linked as reentrant, reusable. CA-7 must be shut down after assembling and link editing this Security module and changing the SECURITY statement in the initialization file. Then, with the startup of CA-7, this security will be in effect. A sample Security module resides on the CA-7 Source library with the name SASSSECI.

7.3.1 Syntax

```
SECURITY
►►──name──SECURITY──APPLID=(xxxx,nn)──,OPID=xxxxxxxx──────────────►

►──,TRM=xxxxxxx──┬─────────────────────┬──┬────────────┬──────────►◄
                 │            ┌─0───┐  │  └─,LAST=YES─┘
                 └─,USRID=────┴─nnn─┴──┘
```

7.3.2 Parameter Descriptions

name
A name, up to 8 characters, specified on the first SECURITY statement coded. It must reflect the CSECT name for the security module. Coding and continuation follow the rules of assembler macro coding.

APLID
Specifies a CA-7 application and the level of functional authority to which the designated operator will be allowed access. APLID is required and has no defaults. Values must be:

xxxx
Identifies the CA-7 application driver (must end in 0). Refer to the Table 7-1 on page 7-4 for each application ID's first three characters.

nn
Indicates the function authorization level with a number from 00 to 15, with 00 being the lowest level and 15 the highest.

Multiple applications may be specified by a sublist in the format:
((XXXX,NN),...,(XXXX,NN))

OPID

Specifies an operator's identification code which must be used when logging on to a CA-7 terminal. Value must be alphanumeric, up to 8 characters. OPID is required and has no default. More than one OPID may be specified in the following format:

OPID=(XXXXXXXX,...,XXXXXXXX)

Multiple OPIDs will assign the same APLID and USRID information to all authorized operators for the terminal specified.

TRM

Specifies the name(s), in up to 7 characters, of the CA-7 terminal(s) which the designated operator(s) are authorized to use. Value must match the NAME value given on the TERM statement in the initialization file defining the terminal. All input terminals defined by TERM statements must be referenced in at least one SECURITY macro statement. TRM is required and there is no default. More than one terminal name can be specified using a sublist notation of TRM=(xxxxxxx,...,xxxxxxx). A value of **ALL** will propagate the specified operator definitions to all terminals defined in the initialization file. At least one TRM=**ALL** must be specified to be able to use the Virtual Terminal feature.

USRID

Specifies a user (ownership) identification which will control the operator's ability to access information in the database. Value must be a number from 0 to 255. USRID is optional. If omitted, the default is zero. USRID=255 allows access to all information regardless of ownership.

LAST

Optional. If used, specifies the last SECURITY statement in the module. The value must be YES.

An assembler END statement must appear after the last SECURITY macro. It is advisable to place a PRINT NOGEN statement before the first SECURITY macro to suppress the macro expansion printout, which can be lengthy.

7.3.3 Example

The following is an example of the Security module:

```
         TITLE 'SECURITY MODULE EXAMPLE'
         PRINT NOGEN
SASSSECA SECURITY TRM=(TERM1,TERM2),OPID=(OP1,OP2),                 X
                 APLID=((SLI0,8),(SPO0,9),(SDM0,3)),                X
                 USRID=23
         SECURITY TRM=(TERM3),OPID=(OP1,OP2,OP3),                   X
                 APLID=((SJR0,10),(SDM0,12)),LAST=YES
         END
```

The following is an explanation of the Security module example:

Terminals TERM1 and TERM2 can be logged on with OPIDs of OP1 or OP2. These terminal names correspond to the TERM statements in the initialization file with NAME=TERM1 and NAME=TERM2. The operators will be able to:

- Enter listing commands (SLI0) requiring an authority level of 8 or less.
- Access jobs (or data sets if the USERID external data set security is in effect) with a UID of 23 or 0 (on the DB.1 screen).
- Enter database maintenance commands (SDM0) requiring an authority level of 3 or less.
- Enter queue maintenance command (SPO0) requiring an authorization level of 9 or less.

Terminal TERM3 can be logged on with OPIDs of OP1, OP2, or OP3. The operators will be able to:

- Perform job restart commands (SJR0) requiring an authority level of 10 or less.
- Perform database maintenance commands (SDM0) requiring an authority level of 12 or less.

7.4 USERID Macro

The USERID macro defines the correspondence, if any, between various UIDs for access to classified jobs, and may also identify data sets external to CA-7 to be protected on a UID level.

To implement this level of security, a USERID Security module must be assembled and link edited (not RENT). It must then be identified on the SECURITY statement, in the initialization file, to allow UID correspondence and data set protection. No sample is provided in the CA-7 Source library since this feature is entirely user dependent.

External data sets not specified in the USERID Security module are not protected by the CA-7 UID feature.

There are two formats associated with the USERID macro. One is for ID correspondence, the other for data set protection. All USERID macros using UID keywords **must** precede the first DSN entry or assembly errors will result.

7.4.1 Syntax

```
 USERID
►►──name──USERID──┬─UID─┬──=──┬─(nnn)──────────────┬──────────────────────►
                  └─U───┘     ├─(nnn,...,nnn)──────┤
                              ├─(nnn-nnn)──────────┤
                              └─(nnn-nnn,,,nnn)────┘

►──┬──────────────────────────────────────────┬──,LAST=YES──────────►◄
   └─,──┬─COIDS─┬──=──┬─(nnn)──────────────┬──┘
        └─C─────┘     ├─(nnn,...,nnn)──────┤
                      ├─(nnn-nnn)──────────┤
                      └─(nnn-nnn,,,nnn)────┘
```

For external data set protection, the format is:

USERID

```
                                    ┌─,──┐
►►──name──USERID──┬─DSNAME─┬─=(──▼xxxx─┴─)──────────────────────────►
                  ├─DSN────┤
                  └─D──────┘

►──┬────────────────────────────────────────┬──────────────────────────►
   └─,─┬─READ─┬─=─┬─(ALL)────────────────┬─┘
       └─R────┘   ├─(nnn)────────────────┤
                  ├─(nnn,...,nnn)────────┤
                  ├─(nnn-nnn)────────────┤
                  ├─(nnn-nnn,,,nnn)──────┤
                  └─(WRITE)──────────────┘

►──┬────────────────────────────────────────┬──┬───────────────────┬──►
   └─,─┬─WRITE─┬─=─┬─(ALL)───────────────┬─┘  └─,MBROPT=─┬─R──┬─┘
       └─W─────┘   ├─(nnn)───────────────┤               ├─W──┤
                   ├─(nnn,...,nnn)───────┤               ├─RW─┤
                   ├─(nnn-nnn)───────────┤               └─WR─┘
                   ├─(nnn-nnn,,,nnn)─────┤
                   └─(READ)──────────────┘

►──,LAST=YES───────────────────────────────────────────────────────►◄
```

7.4.2 Parameter Descriptions

name

Specified only on the first USERID statement in the module to generate the CSECT name for the module.

This parameter is optional and if omitted will default to UIDTABLE.

The macro name USERID should begin in column 10. One space must appear before and after the macro name USERID. Continuation follows the rules of assembler macro coding.

U|UID

Specifies a UID, a range, a list of UIDs, or a list of ranges to have a correspondence with COIDS. UID and U may be used interchangeably. Value may be a single ID, a range of IDs or a sublist combining the previous two values. IDs may be any value between, but not including, 0 and 255 in ascending order.

C|COIDS

Specifies UID(s) that the UID will inherit access to. (This is for job access only, not external data set access, unless MBROPT is used.) May be specified as a UID, range of UIDs, list of UIDs and/or ranges of UIDs in ascending order.

D|DSN|DSNAME

Specifies a data set name or list of data set names to be protected by UID security. DSNAME, DSN and D may be used interchangeably. Value is a data set name or names in up to 44 characters each, which may be enclosed in quotes. A generic name may be indicated by ending it with an asterisk (*) such as SYS*. This keyword may not be specified on a statement containing UID. If DSNAME is used with no other keywords, then access to that data set is only allowed for a UID of 255.

R|READ

Specifies a UID or a list of UIDs or ranges to be allowed read access to data sets in the DSN list in ascending order. READ and R may be used interchangeably. This keyword has the same format as UID. May also be READ=WRITE to indicate that the value is the same as the WRITE list or READ=ALL to indicate unrestricted access. This keyword may not be specified on a statement containing the UID keyword. If MBROPT is used, then READ will be ignored.

W|WRITE

Specifies a UID or list or ranges in ascending order to be allowed write access to data sets in the DSN list. WRITE and W may be used interchangeably. This keyword has the same format as UID. May also be WRITE=READ to indicate that the value is the same as the READ list or WRITE=ALL to indicate unrestricted access. This keyword may not be specified on a statement containing the UID keyword. If MBROPT is used, then WRITE is ignored.

MBROPT= R|W|RW|WR

Specifies member protection for JCL PDS type data sets based on access to like-named jobs in the CA-7 database. May not be used on statements containing the UID keyword. Values may be R for read protection, W for write protection, RW or WR for both read and write protection. If specified, this keyword causes a read of the database for a job having the same name as the PDS member. If a job is found, its UID (and any associated COIDs) control PDS access. If a like-named job is not found, then access is allowed (regardless of READ or WRITE values). This provides a way of extending protection to JCL members for jobs protected with a UID.

LAST=YES

Required. This keyword must be coded only on the last macro statement of the module. Value must be YES.

An assembler END statement must appear after the last USERID macro. It is advisable to place a PRINT NOGEN statement before the first USERID macro to suppress the macro expansion.

7.4.3 Example

The following is an example of the USERID module:

```
         TITLE 'USER-ID SECURITY MODULE'
         PRINT NOGEN
SASSUID  USERID UID=5,COIDS=(7,9,11)
         USERID UID=(10-20),COIDS=(20-25,30)
         USERID DSN=SYS*,WRITE=(200-254)
         USERID DSN=USER.PROCLIB,LAST=YES
         END
```

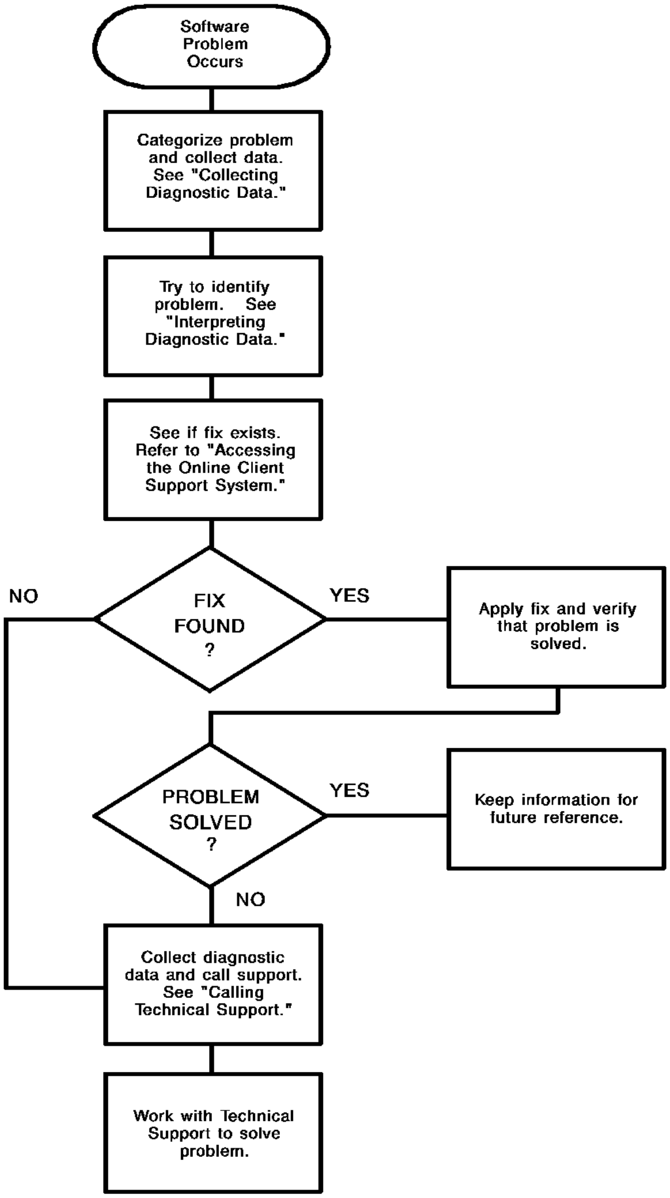
Chapter 8. Troubleshooting

This chapter contains information about:

- Identifying and resolving problems
- Contacting Computer Associates Technical Support
- Receiving ongoing product releases and maintenance
- Requesting product enhancements

8.1 Diagnostic Procedures

Refer to the flowchart below for a summary of the procedures you should follow if you have a problem with a Computer Associates software product. Each of these procedures is detailed on the following pages.


Software Problem Occurs
Categorize problem and collect data. See "Collecting Diagnostic Data."
Try to identify problem. See "Interpreting Diagnostic Data."
See if fix exists. Refer to "Accessing the Online Client Support System."
NO
FIX FOUND ?
YES
Apply fix and verify that problem is solved.
PROBLEM SOLVED ?
YES
Keep information for future reference.
NO
Collect diagnostic data and call support. See "Calling Technical Support."
Work with Technical Support to solve problem.

8.1.1 Collecting Diagnostic Data

The following information is helpful in diagnosing problems that might occur:

- Control statements used to activate your product
- JCL used to install or activate your product
- Relevant system log or console listings
- Relevant system dumps or product dumps
- List of other IBM or third-party products that might be involved
- Manufacturer, model number, and capacity of your hardware
- Numbers and text of IBM or CA error messages associated with the problem
- Names of panels where the problem occurs
- Listings of all fixes applied to all relevant software, including:
  - The dates fixes were applied
  - Fix numbers
  - Names of components to which fixes were applied
- Short description of problems

8.1.2 Interpreting Diagnostic Data

When you have collected the specified diagnostic data, write down your answers to the following questions:

1. What was the sequence of events prior to the error condition?
2. What circumstances existed when the problem occurred and what action did you take?
3. Has this situation occurred before? What was different then?
4. Did the problem occur after a particular PTF was applied or after a new release of the software was installed?
5. Have you recently installed a new release of the operating system?
6. Has the hardware configuration (tape drives, disk drives, and so forth) changed?

From your response to these questions and the diagnostic data, try to identify the cause and resolve the problem.

8.2 Accessing the Online Client Support System

Computer Associates is making extensive use of the Internet for your benefit. CA encourages you to "surf the net" to the CA home page at ***http://www.cai.com***. The CA Internet site provides a great variety of information about CA products and services, including:

- Service and support
- Product information and sales
- CA-World conference information
- Press releases
- CA user groups

CA-TCC (CA-Total Client Care) gives you real time, interactive access to CA product support information through the Internet. Using CA-TCC, you can:

- Open new issues
- Browse or update your existing issues
- Perform keyword searches
- Download solutions, PTFs, and important notices regarding CA products, maintenance, and documentation

8.2.1 Requirements for Using CA-TCC

The following are the requirements to use CA-TCC:

- You must be a CA client with a current maintenance agreement.
- You must register through the CA Internet site.
- You must access the Internet with a browser that supports the HTML specification 2.0 or higher, such as Netscape Navigator 2.0 or higher or Microsoft Internet Explorer 3.0 or higher.

  Browsers that meet the HTML requirement support the following functions, which are required for CA-TCC:

  – Secure sockets layer (SSL) to encrypt your transaction traffic
  – Encrypted data records (known as COOKIES)
  – HTML tables

8.2.2 CA-TCC Security

CA-TCC runs as a secured server (SSL). You may need to configure your browser to enable SSL. Guidelines for doing this are provided on the CA Technical Support page.

8.2.3 Accessing CA-TCC

To access CA-TCC, click on Support on the CA home page and follow the links for CA-TCC. The CA-TCC options are:

- CA-TCC Information
- CA-TCC Registration
- Access CA-TCC

These options are described below.

CA-TCC Information: Select the information option to view background information for CA-TCC, details about the prerequisites, and instructions for configuring your browser. Be sure to review this section for updates or information not included here.

CA-TCC Registration: Select the registration option to identify yourself to CA-TCC. You must register before you can access CA-TCC online. There are prompts for all required information, including your name, site ID, CA-StarTrak PIN, company name, E-Mail address, postal address, and desired password for accessing CA-TCC.

Note: If you do not have a CA-StarTrak PIN, CA-TCC provides one for you when you register.

Access CA-TCC: Select the access option to begin using CA-TCC. When prompted, enter your user ID and password. Once your sign-on is validated, you can select one of the following options:

Open a New Issue
Open an issue for, or request an enhancement to, one of your CA products.

Browse Your Issues
Display all issues for your site. The issues are grouped into three categories: Open, Closed, and Enhancement Requests (DARs).

Browse/Download Solutions
Specify criteria for selecting solutions, which you can then view or download.

Search CA Knowledge Base
Specify criteria for searching the CA database for solutions, problems, and keywords that can provide you with immediate answers to your product support questions and concerns.

Update Your CA-TCC Profile
Make changes to your default E-mail address, phone number, and password whenever necessary.

Display Your Site's Licenses
View a list of all the CA products for which your company site is currently licensed.

Display News Items

View and download recently published solutions for CA products, instructions for downloading from CA-TCC, and helpful information for using CA-StarTrak, CA-TCC, or other CA products.

8.2.4 Accessing the Technical Support Phone Services Directory

The Computer Associates Technical Support Phone Services Directory lists each CA product and the telephone number to call for primary support for that product. To access the Support Phone Services Directory online, click on Support on the CA home page. Follow the links, first to CA Telephone Support and then to the Technical Support Phone Numbers directory.

8.2.5 CA-TCC Hotline

If you experience any problems using CA-TCC, please call the CA-TCC Technical Support hotline at 609-273-3412.

8.3 CA-TLC: Total License Care

Many CA software solutions use license keys or authorization codes to validate your hardware configuration. If you need assistance obtaining a license key or authorization code, contact the CA-TLC: Total License Care group at 1-800-338-6720.

8.4 Calling Technical Support

Computer Associates provides telephone support for all its products.

If you are in North America, refer to the *Technical Support Phone Services Directory* for the appropriate phone number. Outside North America, call your local Computer Associates Support Center during normal business hours.

Note: Only your local Computer Associates Support Center can provide native language assistance. Please use English when contacting any North American center.

If you are unable to locate the Technical Support phone number you need, call 1-800-645-3042 for assistance if you are in North America or 631-342-4683 outside North America. The operator will record your call and a Technical Support representative will call you back. After hours calls should be limited to severity 1 problems.

If you are unable to resolve the problem, please have the following information ready before contacting Computer Associates Technical Support:

- All the diagnostic information described in 8.1.1, “Collecting Diagnostic Data” on page 8-3
- Product name, release number, operating system and genlevel.
- Product name and release number of any other software you suspect is involved.
- Release level and PUTLEVEL of the operating system.
- Your name, telephone number and extension (if any).
- Your company name.
- Your site ID.
- A severity code. This is a number (from 1 to 4) that you assign to the problem. Use the following to determine the severity of the problem:

  1 a "system down" or inoperative condition

  2 a suspected high-impact condition associated with the product

  3 a question concerning product performance or an intermittent low-impact condition associated with the product

  4 a question concerning general product utilization or implementation

8.5 Product Releases and Maintenance

Clients are requested to operate only under currently supported releases of the product.

Clients with current maintenance agreements also receive ongoing product maintenance. When a new release of the system is available, a notice is sent to all current clients.

8.6 Requesting Enhancements

Computer Associates welcomes your suggestions for product enhancements. All suggestions are considered and acknowledged. You can use either of two methods to request enhancements:

- Contact your Account Manager who will initiate a Demand Analysis Request (DAR) for you.
- Enter your request through StarTCC Extended Support on the Web.

Appendix A. CA-7 Security Tables

This appendix includes the following CA-7 security tables:

- CA-7 Panel-ID Table
- CA-7 Command Table
- CA-7 Function and Service Level Table
- CA-7 Access Level Translation Table

A.1 CA-7 Panel-ID Table

The following table lists the CA-7 panel-IDs. Each panel-ID has a corresponding resource name to be used when defining the resource rules for securing CA-7. An asterisk identifies any panel-ID that is new for Version 3.3.

Note: Use the resource name listed in the table for each panel when defining the resource rules under external security. The L2 shown in the resource name is the CA-7 product code and is required.

CA-7 Panel-ID	Resource Name	Description	New in Rel. 3.3
APA	L2AP	CA-7 Automated Performance Analysis Menu (APA)	
AP.1	L2AP1	CA-7 Automated Performance Analysis Prompt	
AP.2	L2AP2	CA-7 Automated Performance Analysis Prompt	
AP.3	L2AP3	CA-7 Automated Performance Analysis Prompt	
AP.4	L2AP4	CA-7 Automated Performance Analysis Prompt	
AP.5	L2AP5	CA-7 Automated Performance Analysis Prompt	
AR.3	L2AR3	CA-7 ARF Condition Definition Maintenance	
DB	L2DB	Database Maintenance Menu (DBM)	
DB.1	L2DB1	DBM - Job Definition	
DB.2	L2DB2	DBM - Scheduling Menu	
DB.2.1	L2DB21	DBM - CPU Job Scheduling	
DB.2.1-E	L2DB21E	DBM - CPU Job Scheduling Parameter Edit	
DB.2.2	L2DB22	DBM - Input Network Scheduling	
DB.2.2-E	L2DB22E	DBM - Input Network Scheduling Parameter Edit	
DB.2.3	L2DB23	DBM - Output Network Scheduling	
DB.2.3-E	L2DB23E	DBM - Output Network Scheduling Parameter Edit	
DB.2.4	L2DB24	DBM - Job Triggering	
DB.2.5	L2DB25	DBM - Input Network Triggering	
DB.2.6	L2DB26	DBM - Data Set Triggering	
DB.2.7	L2DB27	DBM - Modification to Resolve Schedule Dates	
DB.2.8	L2DB28	DBM - Base Calendar Maintenance	
DB.3	L2DB3	DBM - Job Predecessor/Successor Menu	

CA-7 Panel-ID	**Resource Name**	**Description**	**New in Rel. 3.3**
DB.3.1	L2DB31	DBM - Data Set Predecessors	
DB.3.2	L2DB32	DBM - CPU Job Predecessors	
DB.3.4	L2DB34	DBM - Input/Output Network Tasks	
DB.3.6	L2DB36	DBM - User Memo-Form Predecessors	
DB.3.7	L2DB37	DBM - Report-IDs Created	
DB.4	L2DB4	DBM - Workload Documentation Menu	
DB.4.1	L2DB41	DBM - CPU Job Documentation	
DB.4.2	L2DB42	DBM - Input/Output Network Documentation	
DB.4.3	L2DB43	DBM - User-Defined Item Documentation	
DB.4.4	L2DB44	DBM - Data Set Documentation	
DB.4.5	L2DB45	DBM - DD Statement Documentation	
DB.4.6	L2DB46	DBM - Application System Documentation	
DB.5	L2DB5	DBM - Input/Output Network Definition	
DB.6	L2DB6	DBM - Data Set Definition	
DB.7	L2DB7	DBM - JCL Library Maintenance	
DB.8	-na-	DBM -	
QM	L2QM	Queue Maintenance Menu (QM)	
QM.1	L2QM1	QM - CPU Jobs Status Prompt	
QM.1-M	L2QM1M	QM - CPU Jobs Status (RQMTS)	
QM.1-X	L2QM1	QM - CPU Jobs Status	
QM.1-XC	L2QM1C	QM - Reason for Cancel	
QM.1-XE	L2QM5	QM - Queued JCL	
QM.1-XF	L2QM4	QM - CPU Jobs in Restart Status	
QM.1-XH	L2QM1H	QM - CPU Jobs in Hold Status	
QM.1-XJ	L2QM1J	QM - CPU Jobs Status - Reverse JCL Override Requirement	
QM.1-XP	L2QM1P	QM - CPU Jobs Status - Respond to Prompting	
QM.1-XQ	L2QM1Q	QM - CPU Jobs Status - Requeue for a Restart	
QM.1-XR	L2QM1R	QM - CPU Jobs Status - Release from Hold Status	
QM.1-XS	L2QM1S	QM - CPU Jobs Status - Satisfy Submit Time Requirement	
QM.1-XU	L2QM3	QM - CPU Jobs Status - Go to Attribute Update Panel	

CA-7 Panel-ID	Resource Name	Description	New in Rel. 3.3
QM.1-XV	L2QM1V	QM - CPU Jobs Status - Reverse Verify Requirement Status	
QM.1-XX	L2QM2	QM - CPU Jobs Status - Go to Job Predecessor Panel	
QM.2	L2QM2	QM - CPU Job Predecessors Prompt	
QM.2-X	L2QM2	QM - CPU Job Predecessors	
QM.3	L2QM3	QM - CPU Job Attributes Prompt	
QM.3-X	L2QM3	QM - CPU Job Attributes	
QM.4	L2QM4	QM - CPU Job in Restart Status Prompt	
QM.4-X	L2QM4	QM - CPU Job in Restart Status	
QM.5	L2QM5	QM - Queued JCL	
QM.6	L2QM6	QM - Input Networks Prompt	
QM.6-S	L2QM6	QM - Input Networks (2 Up Display)	
QM.6-SC	L2QM6C	QM - Input Networks - Cancel (2 Up Display)	
QM.6-SF	L2QM6F	QM - Input Networks - Force (2 Up Display)	
QM.6-SH	L2QM6H	QM - Input Networks - Hold (2 Up Display)	
QM.6-SI	L2QM6I	QM - Input Networks - Login (2 Up Display)	
QM.6-SO	L2QM6O	QM - Input Networks - Logout (2 Up Display)	
QM.6-SP	L2QM6P	QM - Input Networks - Respond to Prompting (2 Up Display)	
QM.6-SR	L2QM6R	QM - Input Networks - Release from Hold (2 Up Display)	
QM.6-X	L2QM6	QM - Input Networks	
QM.6-XC	L2QM6C	QM - Input Networks - Cancel	
QM.6XF	L2QM6F	QM - Input Networks - Free	
QM.6XH	L2QM6H	QM - Input Networks - Hold	
QM.6XI	L2QM6I	QM - Input Networks - Login	
QM.6XO	L2QM6O	QM - Input Networks - Logout	
QM.6XP	L2QM6P	QM - Input Networks - Respond to Prompting	
QM.6XR	L2QM6R	QM - Input Networks - Release from Hold	
QM.7	L2QM7	QM - Output Networks Prompts	
QM.7-S	L2QM7	QM - Output Networks (2 Up Display)	
QM.7-SC	L2QM7C	QM - Output Networks - Cancel (2 Up Display)	
QM.7-SF	L2QM7F	QM - Output Networks - Force (2 Up Display)	

CA-7 Panel-ID	**Resource Name**	**Description**	**New in Rel. 3.3**
QM.7-SH	L2QM7H	QM - Output Networks - Hold (2 Up Display)	
QM.7-SI	L2QM7I	QM - Output Networks - Login (2 Up Display)	
QM.7-SO	L2QM7O	QM - Output Network - Logout (2 Up Display)	
QM.7-SP	L2QM7P	QM - Output Networks - Respond to Prompting (2 Up Display)	
QM.7-SR	L2QM7R	QM - Output Networks - Release from Hold (2 Up Display)	
QM.7-X	L2QM7	QM - Output Networks	
QM.7-XC	L2QM7C	QM - Output Networks - Cancel	
QM.7-XF	L2QM7F	QM - Output Networks - Free	
QM.7-XH	L2QM7H	QM - Output Networks - Hold	
QM.7-XI	L2QM7I	QM - Output Networks - Login	
QM.7-XO	L2QM7O	QM - Output Networks - Logout	
QM.7-XP	L2QM7P	QM - Output Networks - Respond to Prompting	
QM.7-XR	L2QM7R	QM - Output Networks - Release from Hold	
RM	L2RM	Virtual Resource Management Menu (RM)	
RM.1	L2RM1	RM - Job Resource Management	
RM.2	L2RM2	RM - Resources and Jobs Cross Reference List	
RM.3	L2RM3	RM - Active Job Resources Display	
RM.4	L2RM4	RM - Pending Resources Job Display	
RM.5	L2RM5	RM - Jobs Waiting on Resources	
RM.6	L2RM6	RM - Corequisite Resources List	
RM.7	L2RM7	RM - Resource Count Management	
UT	L2UT	CA-7 Utilities	
UT.1	L2UT1	CA-7 Utilities - Allocate Data Set	
UT.1-C	L2UT1C	CA-7 Utilities - Allocate/Catalog Data Set	
UT.10	L2UT10	CA-7 Utilities - Find Data Set on DASD	
UT.11	L2UT11	CA-7 Utilities - Allocate Volume	
UT.12	L2UT12	CA-7 Utilities - Deallocate Volume	
UT.13	L2UT13	CA-7 Utilities - Display Format 1 DSCB	
UT.14	L2UT14	CA-7 Utilities - Display Directory Information	
UT.15	L2UT15	CA-7 Utilities - Display Data Set Attributes Map	

CA-7 Panel-ID	**Resource Name**	**Description**	**New in Rel. 3.3**
UT.16	L2UT16	CA-7 Utilities - Display Available DASD Space	
UT.17	L2UT17	CA-7 Utilities - Display Physical Data Records	
UT.18	L2UT18	CA-7 Utilities - Display Catalog Block	
UT.19	L2UT19	CA-7 Utilities - Display Catalog Entries	
UT.2	L2UT2	CA-7 Utilities - Catalog Data Set	
UT.3	L2UT3	CA-7 Utilities - Rename Data Set	
UT.4	L2UT4	CA-7 Utilities - Scratch Data Set	
UT.4P	L2UT4P	CA-7 Utilities	
UT.5	L2UT5	CA-7 Utilities - Uncatalog Data Set	
UT.6	L2UT6	CA-7 Utilities - Build GDG Index	
UT.7	L2UT7	CA-7 Utilities - Delete Index	
UT.8	L2UT8	CA-7 Utilities - Connect a Catalog	
UT.9	L2UT9	CA-7 Utilities - Disconnect a Catalog	
WB.X	L2WBX	CA-7 Workload Balancing Maintenance	

A.2 CA-7 Command Table

The following table lists the CA-7 commands and the corresponding resource name to be used when defining the resource rules to external security. The service level is READ. The prefix of L2 on the resource names is the CA-7 product code and is required. The command entries that have a resource name of n/a (not applicable) do not require access authorization. These commands only affect the issuing user's current terminal environment. Refer to the *CA-7 Commands Guide* for a detailed description of the commands listed in this table.

CA-7 Command	Resource Name	New in Version 3.3
/ASSIGN	L2SCASSI	
/AUTO	L2SCAUTO	
/BRO	L2SCBRO	
/CHANGE	L2SCCHAN	
/CLOSE	n/a	
/CLOSE(T)	L2SCCLOS	
/COPY	n/a	
/DISPLAY	n/a	
/DMP1	L2SCDMP1	
/DUMP	L2SCDUMP	
/ECHO	n/a	
/FETCH	n/a	*
/JCL	L2SCJCL	
/LOG	L2SCLOG	
/LOGOFF	n/a	
/LOGOFF(T)	L2SCLGOF	
/LOGON	n/a	
/MSG	L2SCMSG	
/MVS	L2SCMVS	
/NXTMSG	n/a	
/OPEN	n/a	
/OPEN(T)	L2SCOPEN	

CA-7 Command	Resource Name	New in Version 3.3
/OPERID	L2SCOPER	
/OPERIDS	L2SCOPRS	
/PA	n/a	
/PAGE	n/a	
/PF	n/a	
/PROF	n/a	
/PROFS	L2SCPROF	
/PURGPG	n/a	
/PURGPG(T)	L2SCPURG	
/REFRESH	L2SCRFSH	
/RELINK	L2SCRLNK	
/RESET	L2SCRSET	
/SHUTDOWN	L2SCSHUT	
/START	L2SCSTAR	
/STOP	L2SCSTOP	
/SWAP	L2SCSWAP	
/UID	L2SCUID	
/WLB	L2WBSWLB	
/WTO	L2SCWTO	
ADDRQ	L2QPADRQ	
ADDSCH	L2QPADSC	
AL	L2UT1	
ALC	L2UT1C	
ALLOC	L2UT11	
ARFP	L2QPARFP	
ARTS	L2CAARTS	
BLDG	L2UT6	
CALMOD	L2DB28	
CANCEL	L2QPCNCL	
CAT	L2UT2	

CA-7 Command	Resource Name	New in Version 3.3
CLEAR	n/a	
CONN	L2UT8	
CTLG	L2QPCTLG	
DCONN	L2UT9	
DEALLOC	L2UT12	
DEMAND	L2QPDMND	
DEMANDH	L2QPDMND	
DIRECT	L2QPDREC	
DLTX	L2UT7	
DM	L2TSDM	
DMDNW	L2QPDMNW	
DMP	L2TSDMP	
DMPCAT	L2UT18	
DMPDSCB	L2UT13	
DMPDSN	L2UT17	
DSN	L2DB6	
DUMP	L2GIDUMP	
EDIT	n/a	
FALL	L2FCFALL	
FIND	L2UT10	
FIX	L2TSFIX	
FJOB	L2FCFJOB	
FLOWD	L2QPFLWD	*
FLOWL	L2GIFLWL	*
FPOST	L2FCFPOS	
FPRE	L2FCFPRE	
FQALL	L2FCFQAL	
FQJOB	L2FCFQJO	
FQPOST	L2FCFQPO	
FQPRE	L2FCFQPR	

CA-7 Command	Resource Name	New in Version 3.3
FQRES	L2FCFQRE	
FQSTN	L2FCFQST	
FQTAPE	L2FCFQTA	
FRE	L2TSFRE	
FRES	L2FCFRES	
FRJOB	L2FCFRJO	
FRQJOB	L2FCFRQJ	
FSTN	L2FCFSTN	
FSTRUC	L2FCFSTR	
FTAPE	L2FCFTAP	
FWLP	L2FCFWLP	
GO	L2TSGO	
GRAPHD	L2AP3	
GRAPHJ	L2AP1	
GRAPHN	L2AP4	
GRAPHS	L2AP2	
HELP	n/a	
HOLD	L2QPHOLD	
IN	L2QPIN	
IO	L2QPIO	
JCL	L2DB7	
JCLOVRD	L2QPJCLO	
JOB	L2DB1	
JOBCONN	L2DB3	
LACT	L2GILACT	
LACTR	L2GILACR	
LARF	L2GILARF	
LARFQ	L2GILARQ	
LCTLG	L2GILCTL	
LDSN	L2GILDSN	

CA-7 Command	Resource Name	New in Version 3.3
LDTM	L2GILDTM	
LIST	L2GILIST	
LISTDIR	L2UT14	
LJCK	L2GILJCK	
LJCL	L2GILJCL	
LJES	L2GILJES	
LJOB	L2GILJOB	
LJOBR	L2GILJOR	
LLIB	L2GILLIB	
LLOCK	L2GILLOC	
LNTWK	L2GILNWK	
LOAD	L2QPLOAD	
LOADH	L2QPLOAD	
LOC	L2UT19	
LOGIN	L2QPLGIN	
LOGOUT	L2QPLGOU	
LPDS	L2GILPDS	
LPOST	L2GILPOS	
LPRE	L2GILPRE	
LPROS	L2GILPRO	
LPRRN	L2GILPRN	
LQ	L2GILQ	
LQP	L2GILQP	
LQUE	L2GILQ	
LQR	L2GILQR	
LRDY	L2GILRDY	
LRDYP	L2GILRDP	
LRDYR	L2GILRDR	
LREQ	L2GILREQ	
LREQP	L2GILREP	

CA-7 Command	Resource Name	New in Version 3.3
LREQR	L2GILRER	
LRES	L2GILRES	
LRLOG	L2GILRLO	
LRMD	L2GILRMD	
LSCHD	L2GILSCH	
LSYS	L2GILSYS	
LTR	L2TSLTR	
LVAR	L2GILVAR	
LWLB	L2WBLWLB	
MAP	L2UT15	
MENU	n/a	
MOVE	L2TSMOVE	
NETWORK	L2DB5	
NOPRMP	L2QPNOPR	
NXTCYC	L2QPNXTC	
OUT	L2QPOUT	
PAT	L2TSPAT	
POST	L2QPPOST	
PRINT	L2GIPRNT	
PRMP	L2QPPRMP	
PROSE	L2DB4	
PRRNJCL	L2QPPRNJ	
PRSCF	L2QPPRCF	
PRSQA	L2QPPRQA	
PRSQD	L2QPPRQD	
PS	L2PS	
QJCL	L2QM5	
RELEASE	L2QPRLSE	
REMIND	L2QPRMIN	
RENAME	L2UT3	

CA-7 Command	Resource Name	New in Version 3.3
REQUEUE	L2QPRQUE	
RESANL	L2DBRSNL	
RESCHNG	L2QPRSCH	
RESOLV	L2DBRSLV	
RESTART	L2QPREST	
RQMT	L2DBRQMT	
RQVER	L2QPRQVR	
RSVP	L2QPRSVP	
RUN	L2QPRUN	
RUNH	L2QPRUN	
RUNNW	L2QPRNNW	
RUSH	L2QPRUSH	
SAV	L2TSSAV	
SCHD	L2DB2	
SCHDMOD	L2DB27	
SCRATCH	L2UT4	
SCRATCHP	L2UT4P	
SPACE	L2UT16	
SSCAN	L2SCSCAN	
START	L2QPSTAR	
STOP	L2QPSTOP	
SUBMIT	L2QPSUBM	
SUBSCH	L2QPSUBS	
SUBTM	L2QPSUBT	
SYSDMP	L2TSSYSD	
SYSINQ	L2TSSYSI	
TIQ	L2CATIQ	
TIQU	L2CATIQU	
TRA	L2TSTRA	
TRIG	L2DBTRIG	

CA-7 Command	**Resource Name**	**New in Version 3.3**
TRP	L2TSTRP	
UNC	L2UT5	
UT*	L2UT	
VERIFY	L2QPVERI	
X	L2TSX	
XPOST	L2QM7	
XPRE	L2QM6	
XQ	L2QM1	
XQJ	L2QM1	
XQM	L2QM1M	
XQN	L2QM1	
XREF	L2DBXREF	
XRQ	L2QM2	
XRST	L2QM4	
XSPOST	L2QM7S	
XSPRE	L2QM6S	
XUPD	L2QM3	
XWLB	L2WBX	
ZA	L2TSZA	
ZAP	L2TSZAP	

Note: Access to the commands with a resource name prefix of L2TS must be restricted. These are diagnostic commands generally only issued at the request of CA-7 Technical Support personnel. The ability to display and modify storage with these commands could allow unauthorized access of sensitive data.

A.3 CA-7 Function and Service Level Table

The function table lists the CA-7 functions and a corresponding service level required to perform that function. Each function may have additional aliases. The service level must be specified on the resource rule for a given panel or command to grant access to that function. Service levels are translated for external security calls according to A.4, “CA-7 Access Level Translation Table” on page A-17.

Function(s)	Service Level	Alias
LIST	READ	L,LDD,LDIT,LISTA,LISTP,LISTR,LPD
FETCH	READ	F
FE	READ	FEIT,FEPL,FEVE
SAVE	ADD	S
REPL	UPDATE	R,REP
EDIT	n/a	E,EDITH
EXIT	n/a	n/a
UPD	UPDATE	U,UDD,UIST,UPDATE,UPDT
ADD	ADD	A,ADDT,AELETE,AIST,APD
DELETE	DELETE	D,DEL,DELT
DD	DELETE	n/a
FREE	DELETE	n/a
DELPRRN	UPDATE	n/a
FETCHP	READ	FP
FPE	READ	n/a
APPEND	READ	AP,APP
APPENDP	READ	n/a
RENAME	UPDATE	REN
RUN	SUBMIT	n/a
RUNH	SUBMIT	n/a
SR	UPDATE	n/a
SS	ADD	n/a
SUBMIT	SUBMIT	SUB
RESOLV	SUBMIT	RES

Function(s)	**Service Level**	**Alias**
CLEAR	n/a	CL,CLR
FORMAT	n/a	FMT,FOR,FORM
XPRE	UPDATE	n/a
XRST	UPDATE	n/a
XSPRE	UPDATE	n/a
XPOST	UPDATE	n/a
XSPOST	UPDATE	n/a
XQ	UPDATE	n/a
XQJ	UPDATE	n/a
XQM	UPDATE	n/a
XQN	UPDATE	n/a
XRQ	UPDATE	n/a
XUPD	UPDATE	n/a
XWLP	UPDATE	n/a
REQ	UPDATE	n/a
RET	SUBMIT	n/a

A.4 CA-7 Access Level Translation Table

The following table describes the access levels for CA-7 commands and panels and how they are translated by the CA Standard Security Facility (CAISSF). Refer to the appropriate column for the security package that you have implemented at your installation for the equivalent access level to specify when defining access authorization for CA-7 users.

CA-7	**CAISSF**	**CA-Top Secret**	**CA-ACF2**	**RACF**
Read	Read	Read	Read	Read
Update	Update	Update	Update	Update
Add	Create	Create	Add	Control
Delete	Scratch	Scratch	Delete	Control
Submit	Control	Control	Update	Control

Index