CA Workload Automation CA 7® Edition

Security Reference Guide

Version 12.0.00

technologies

This Documentation, which includes embedded help systems and electronically distributed materials (hereinafter referred to as the “Documentation”), is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2013 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

CA Technologies Product References

This document references the following CA Technologies products:

- CA Workload Automation CA 7® Edition, (CA WA CA 7 Edition), formerly CA Workload Automation SE and CA 7® Workload Automation
- CA ACF2™
- CA Dispatch™
- CA Endevor® Change Manager (CA Endevor)
- CA Top Secret®
- CA NSM Job Management Option (CA NSM JMO)
- CA Workload Automation AE, formerly CA AutoSys®
- CA Workload Automation Restart Option for z/OS Schedulers (CA WA Restart Option), formerly CA 11™ Workload Automation Restart and Tracking

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 6: Implementing Security with IBM RACF 105

Chapter 1: Introduction to Security

This guide describes the steps necessary to implement CA Workload Automation CA 7 ® Edition (CA WA CA 7 Edition) security.

Note: Because this document contains sensitive information, we recommend that you limit the distribution of this guide. Give access to those employees responsible for implementation and maintenance of CA WA CA 7 Edition security.

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

- External security interface
- CA WA CA 7 Edition internal security

Use of external security to control access to CA WA CA 7 Edition resources lets you centralize maintenance of security. If the external security interface is used, you can extend existing USERID definitions and data set access rules for use in the CA WA CA 7 Edition environment.

CA WA CA 7 Edition internal or "native" security can control data and resources that are unique to the CA WA CA 7 Edition processing environment.

Chapter 2: Implementation Considerations

Review these topics to become familiar with CA WA CA 7 Edition security.

This section contains the following topics:

System Requirements

CA WA CA 7 Edition interfaces with your external security product.

The external security interface requires CA Common Services CAIRIM and CAISSF at a currently supported level.

Your external security product can be CA ACF2, CA Top Secret, or RACF. Your external security product must operate at a currently supported level.

Identify Your Security Requirements

The user can define the security structure and level of authority for each individual accessing CA WA CA 7 Edition. Make a careful evaluation of your security requirements before implementing CA WA CA 7 Edition 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. An authorization level must be defined to protect your data and resources. CA WA CA 7 Edition provides facilities for accessing and maintaining the database and monitoring and controlling the production process. These facilities require performance of specialized tasks by different people with varying levels of responsibility.

To help ensure integrity of its system, CA WA CA 7 Edition lets each installation define a control structure or hierarchy required within its environment. Levels of responsibility are then assigned to specific individuals depending on that individuals activities within the hierarchy.

When defining the control structure, consider the following:

- Which security options are you using?
- Who can access and use CA WA CA 7 Edition?
- Do individual users need access to only certain commands or applications under CA WA CA 7 Edition?
- On formatted panels, are users be restricted from certain functions (Add, Update, Delete)?
- Can users access calendars?
- Do USERIDs exist in the JCL for a job or does CA WA CA 7 Edition or a user exit insert the USERIDs?
- Can users issue commands or submit jobs that contain a USERID other than their own?

Security Structure Using External Security

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

- Logon
- Data Set
- Command Authority
- Panel Access
- Calendar Access
- Job Submission
- External Communicators
- USERID Protection
- Multi-CPU Security Environments
- CA WA CA 7 Edition 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 security definitions in place at the time of the access determine the authority of the user to access the data or resource. For a description of the control points, see the following topics.

Logon

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

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

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

Data Set Security

Data set security is used to control access to data sets by users signed on to CA WA CA 7 Edition. Security validates each attempt to access data through CA WA CA 7 Edition. Access authorization to data sets through CA WA CA 7 Edition should not differ from the access you have defined for users outside of the CA WA CA 7 Edition environment. Access is done under the CA WA CA 7 Edition USERID.

Panel Access

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

Calendar Access

Calendar security is provided to control access to calendars by users signed on to CA WA CA 7 Edition. Security validates each attempt to access calendars through CA WA CA 7 Edition. Access is done under the CA WA CA 7 Edition USERID.

Job Submission

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

External Communicators

The External Communicators provide a means for users outside the CA WA CA 7 Edition address space to send terminal transactions or post data set creations to CA WA CA 7 Edition. The following are the External Communicators:

- SASSBSTR
- SASSTRLR
- U7SVC
- SASSBCLP

For more information about the secure use of the External Communicators, see the chapters of this guide that specifically target implementation concerns for your security environment (that is, CA ACF2, CA Top Secret, or IBM RACF).

Command Authority

Command security lets you control access to the wide range of command options available under CA WA CA 7 Edition. Because CA WA CA 7 Edition offers some powerful commands, it is important that you restrict the command authority for each user of CA WA CA 7 Edition to individual areas of responsibility.

USERID Protection

In the standard security environment, each user is assigned a USERID that restricts the users' access to resources related to their 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:

- Requests for jobs, such as DEMANDs, LOADs, or RUNs when the job's JCL contains a USERID.
- Attempts to add USERIDs to JCL through the CA WA CA 7 Edition text editor or queue JCL editor.
- Attempts to add, update, or delete the OWNER field associated with a job on the job definition panel.
- Executions of the SASSTRLR or SASSBSTR facilities when a USERID is supplied on a /LOGON statement. 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.

UID Assignment

External security using UID Resources can now control the UID assignment for users of CA WA CA 7 Edition. A UID Resource Table (default - SASSRTBL) lets sites define a resource name to a UID value relationship. The name can be validated through external security during logons to CA WA CA 7 Edition. This validation eliminates the need to maintain the CA WA CA 7 Edition internal security module with all USERIDs. This validation provides the UID level security assignment that external security controls. For more information about implementing UID Resources, see the section on the appropriate security package.

Multi-CPU Security Environments

CA WA CA 7 Edition 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 WA CA 7 Edition does not support multiple security environments for remote job submission due to the JCL USERID format restrictions imposed by the individual security packages.

CA 7 Submit Data Sets

The 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 WA CA 7 Edition. ICOM reads it for submission to the appropriate system. With USERID insertion during job submission, the JCL USERID format is determined based on the external security environment that CA WA CA 7 Edition identified during initialization. Due to JCL USERID format restrictions related to each external security package, CA WA CA 7 Edition does not support multiple security environments for USERID insertion.

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

Security Structure Using Internal Security

You can define five distinct levels of security using CA WA CA 7 Edition internal security:

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

More information:

Internal Security (see page 139)

Security Considerations for ARF

ARF allows automation of customized recovery procedures for production jobs. The recovery procedures for an ARF condition can include the following:

- Sending a message to a specified TSO user or to the MVS console.
- Executing CA WA CA 7 Edition commands
- Scheduling and tracking special recovery (ARFJ) jobs.

Note: For more information about ARF, see the *Database Maintenance Guide*.

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

Each recovery action statement (except AW statements) can cause the execution of one or more CA WA CA 7 Edition terminal commands. Although the format of the command must follow conventions used for batch terminal or SASSTRLR input, ARF responses use neither of these terminal types. Instead, ARF requires that one or more terminals be defined in the 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 WA CA 7 Edition such as ARF.

Because ARF recovery actions are processed as CA WA CA 7 Edition 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 requires 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 initialization file, the ID must be valid for ALL of these terminals because ARF transactions can be scheduled on any of these terminals.

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

Security Considerations for Cross-Platform Scheduling

CA WA CA 7 Edition can send and receive job requests to and from other CA scheduling systems on various platforms. Cross-platform scheduling is documented in the *Interface Reference Guide*. This section reviews the security concerns of cross-platform scheduling and directs you to relevant documentation in this and other CA WA CA 7 Edition guides.

Three forms of cross-platform scheduling are supported:

- Submit jobs defined using the AGJOB command to a CA system agent directly.
- Submit jobs defined using the XPJOB command to a CA scheduling system or agent directly.
- Submit jobs using the CA7TOUNI batch program.

The security concerns for each are different.

Note: For more information about each mode of submission, see the *Interface Reference Guide*.

CA 7 As Cross-Platform Client

CA WA CA 7 Edition associates two different types of user IDs with agent jobs.

The first type is the agent user ID. An agent user ID is sent to the agent for authorization on the agent platform. Most agent job types require one agent user ID, although some job types require more, and some do not require any. The agent user IDs come from either the job definition in CA WA CA 7 Edition or from the CLANG statements in the PARMLIB data. Passwords associated with user IDs, agents, and job types are maintained through the AGPSWD command.

The second type of user ID associated with an agent job is the mainframe user ID (MFUser). Every agent job that CA WA CA 7 Edition submitts requires an MFUser field. When internal security is used or when external security is used but agent job submissions are not being validated, the CA7 global user ID is used as the MFUser field. When external security is used and agent job submissions are being validated, CA WA CA 7 Edition performs a hierarchical search to determine the MFUser field. The AGUSER keyword on the SECURITY statement in the initialization file determines the search path. For commands, MFUser is the currently signed-on user who is issuing the command. '

The following are the choices:

OWNER

Indicates to select the job OWNER ID as the MFUser field.

REQ

Indicates the requester's ID is a candidate for the MFUser field. The requester ID can be the user ID of a user issuing a DEMAND command to request a job. The requester ID can also be the user ID selected for a job (requester) that then triggers additional jobs. The triggered jobs inherit the user ID. For data set triggers, jobs that create or "post" a data set to CA WA CA 7 Edition have their associated user ID propagated to any later triggered jobs. For the U7SVC and SASSBCLP facilities, the user ID is extracted from the current environment from which the user issues the data set creation or post request.

QJCL

Indicates that the user ID of any user editing queue PARMLIB data for a job is a candidate for the MFUser field. This value would be the user ID of the last person to edit queue PARMLIB data for a job.

CA7

Indicates to select the user ID assigned to CA WA CA 7 Edition for the MFUser field.

If validating agent job submission, as determined by the presence of EXTERNAL=(…AGENT) in the SECURITY initialization statement, authorizations are performed to help ensure that MFUser is authorized to submit agent jobs to the specific agent name using the agent user ID. Authorizations are also performed for commands directed to an agent. The AGLCASS keyword on the SECURITY statement determines the resource class used for the authorizations.

Note: For more information, see the SECURITY statement in this guide and the AGPSWD command in the *Database Maintenance Guide*.

CA 7 as Cross-Platform Client

Using XPJOB definitions, sites can take advantage of the hierarchical security designed especially for these jobs. The XPDEF file initialization statement PSWDLOC keyword defines the four supported security modes. The following are the modes:

DATABASE

Creates an owner security record in the database using the XPSWD command. User ID, password, and domain information can be associated to the owner. Any XPJOB definition containing the same owner uses the information in this record when building the request to transmit to the remote operating environment. Password information provided using the XPSWD command is encrypted and nondisplayable.

OWNER

Uses the OWNER field in the XPJOB definition as the user ID that is passed to the remote node. No password or domain information is provided.

NODE

Creates a node security record in the database using the XPSWD command. User ID, password, and domain information can be associated to the node. Any XPJOB definition containing the same node uses the information in this record when building the request to transmit to the remote operating environment. Password information provided using the XPSWD command is encrypted and nondisplayable.

USER

Supplies SUBUSER and SUBPASS information in an external file associated to the XPJOB definition. You can use the standard facilities of your MVS security system to secure this external file for READ and WRITE access.

The DATABASE and NODE modes are the most secure. In addition, they provide an alternative to supplying a user ID and password to each job definition. With these two modes, all XPJOBs whose OWNER or NODE match a DATABASE or NODE security record use the information in that security record.

Note: For more information about the XPDEF PSWDLOC statement, see the *Systems Programming Guide*. For more information about the XPSWD command, see the *Database Maintenance Guide*.

If you want to supply the user ID of ROOT, create an XPSWD Owner or Node security record. In addition, the SUBROOT keyword of the XPDEF file initialization statement must be set to Y.

XPJOB definitions are defined, scheduled, and/or demanded in the same manner as any other CA WA CA 7 Edition batch job. A user ID is always passed to the target system with password and domain being optional. The source of this information is dependent upon the XPDEF PSWDLOC file initialization statement definition.

CA 7 As Cross-Platform Client

The primary security considerations for CA7TOUNI cross-platform job definitions relate to the CA WA CA 7 Edition SUBMIT function.

- The MVS USERID under which the CA WA CA 7 Edition cross-platform tracking function runs requires both READ and UPDATE access to the CA WA CA 7 Edition XPS PROFILE partitioned data set. The USERID creates and updates a member for each remote system to which cross-platform requests are sent.
- All MVS USERIDs under which the CA WA CA 7 Edition cross-platform submit function runs must have READ access to the CA WA CA 7 Edition 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 the data set.
- CA WA CA 7 Edition cross-platform submit jobs are defined, scheduled, and/or demanded in the same manner as any other CA WA CA 7 Edition batch job. The MVS USERID under which the batch submit job runs is assigned in the same manner as your other CA WA CA 7 Edition batch jobs.
- 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.

- A security call can be made to the external security system to determine whether 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 validation is the same Submit Check that can be done in CA WA CA 7 Edition (see 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:

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

    - and -

  - Either the value of BSUBCHK for this instance is Y or the SUBCHECK=YES parameter is set in the CA WA CA 7 Edition XPS PROFILE member CACCENV. The BSUBCHK value controls CA WA CA 7 Edition security checking for processes outside of the CA WA CA 7 Edition address space (BTI, U7SVC, and so forth). The SUBCHECK= parameter cannot suppress Submit Checking when the value of the BSUBCHK is Y.

    CAIRIM sets the value of BSUBCHK for each instance. For more information, see the chapter "Execution" in the *Systems Programming Guide*.

    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. The batch Submit job fails.

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

- The target system sometimes requires 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 that 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.

CA 7 As Cross-Platform Server

When CA WA CA 7 Edition receives a cross-platform request from a CA scheduling system on another operating environment, it acts as a cross-platform server. This process is fully documented in the *Interface Reference Guide*. The primary security considerations relate to the assignment of a CA WA CA 7 Edition USERID under which the requested CA WA CA 7 Edition job is initiated.

- Cross-platform requests sent to CA WA CA 7 Edition do not always have an explicit USERID sent with them. If an explicit USERID is sent, it does not always have an associated password sent with it.

  If a USERID is sent with a request, you can require that a password is also sent based on the system that sent the request, the USERID, or both that is specified on the request. See the subsection Cross-Platform Server Password Requirements in the *Interface Reference Guide* for documentation about 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 WA CA 7 Edition itself.

- If no USERID is sent with a request, it can be assigned a default CA WA CA 7 Edition USERID based upon the setting of the XPSSID= keyword on the CA WA CA 7 Edition SECURITY statement. For more information, see the SECURITY statement.

- Once a cross-platform request is passed to CA WA CA 7 Edition, processing takes place based upon the explicit or defaulted CA WA CA 7 Edition USERID. This USERID is 'logged on to' a CA WA CA 7 Edition 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 WA CA 7 Edition system. That is, it is handled using internal or external security based upon the global parameters you have specified on your CA WA CA 7 Edition SECURITY statement. If the logon is successful, the CA WA CA 7 Edition 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.

Security Choice

The implementation and structure of CA WA CA 7 Edition security differs based on your choice of internal or external security. For more information about the various security options to determine which options meet your installation's security requirements, see the appropriate chapter in this guide.

Chapter 3: Security Initialization Options

You can set security options using the SECURITY statement in the initialization file. You can check the options in effect by using the /DISPLAY command.

This section contains the following topics:

SECURITY Statement

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

This statement has the following format:

```
SECURITY,NAME=SASSSECI
    [,ACF2CARD={JOBFROM|LOGONID}]
    [,AGCLASS={FACILITY|xxxxxxxx}]
    [,AGUSER=(OWNER,REQ,QJCL,CA7)]
    [,APPL=CA7]
    [,BYPSEC=(1,2,3)]
    [,CCLASS={CALENDAR[1]|xxxxxxxx}]
    [,DFLTUSER=xxxxxxxx]
    [,DISPLAY={YES|NO}]
    [,EXTERNAL=(AGENT,CALENDAR,COMMAND,DATASET,LOGON,SUBCHECK,SUBOWNER)]
    [,HIDEGRP={NO|YES}]
    [,HIDEPW={NO|YES}]
    [,HIDEUPD={NO|YES}]
    [,HIDEUSER={NO|YES}]
    [,JCLUID={YES|NO}]
    [,LOADSUBC={YES|NO}]
    [,LOGOPID={YES|NO|ALL}]
    [,MIXPW={NO|YES}]
    [,MULTIJOB={IGNORE|FLUSH|REQUEUE}]
    [,PCLASS=xxxxxxxx]
    [,PROPAGATE={NO|YES}]
    [,PSOWNER={YES|NO}]
    [,RCLASS=xxxxxxxx]
    [,RESLOGON={YES|NO}]
    [,RLOGUID={YES|NO}]
    [,SCLASS={SUBMIT|xxxxxxxx}]
    [,STATSID={YES|NO}]
    [,SUBNOID={NO|YES}]
    [,SUBUID=(OWNER,REQ,QJCL,DEFAULT,CA7)]
    [,UID=xxxxxxxx]
    [,USER=xxxxxxxx]
    [,XBSCLASS={YES|NO}]
    [,XBSUBCHK={YES|NO}]
    [,XPSSID={**NONE**|xxxxxxxx}]
```

[1] For CA Top Secret, the default is $CALNDR because CALENDAR is a reserved word in CA Top Secret.

SECURITY

Identifies the SECURITY statement that describes the CA WA CA 7 Edition security environment.

NAME

Identifies the load module that contains the 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 WA CA 7 Edition, the default security module SASSSECI can be used. This module is supplied in the CA WA CA 7 Edition load library. If you are using CA WA CA 7 Edition internal security, see the internal security chapter for more information about building and modifying the CA WA CA 7 Edition security module to meet your installation's security requirements.

ACF2CARD

(Optional) For USERID insertion at CA ACF2 sites, CA WA CA 7 Edition adds a control statement to the JCL immediately following the JOB statement. This option controls what type of CA ACF2 control statement is used. JOBFROM is the default. The only other valid value is LOGONID. For more information, see the JCL USERID Format topic.

AGCLASS

(Optional) Specifies the resource class used for security calls that are made from CA WA CA 7 Edition to validate a user's authority to submit agent jobs or execute agent commands. The default resource class is FACILITY. This field can be up to eight characters.

AGUSER

(Optional) Specifies a hierarchy of candidate user ID sources to determine the mainframe user (MFUser) to use for authorizing job submission for agent jobs. This list prioritizes the potential sources for user IDs. The order of specification in the list determines the priority of the user ID to select. If you are validating agent job submissions, authorizations are performed to verify that MFUser is authorized to submit agent jobs to the specific agent name using the agent user ID. The AGCLASS keyword determines the resource class used for the authorization.

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

OWNER

Indicates to select the job OWNER ID as the MFUser field.

REQ

Indicates the requester ID is a candidate for the MFUser field. The requester ID can be the user ID of a user issuing a DEMAND command to request a job. The requester ID can also be the user ID selected for a job (requester) that then triggers additional jobs. The triggered jobs inherit the user ID. For data set triggers, jobs that create or "post" a data set to CA WA CA 7 Edition have their associated user ID propagated to any later triggered jobs. For the U7SVC and CA WA CA 7 Edition SASSBCLP facilities, the user ID is extracted from the current environment from which the user issues the data set creation or post request.

QJCL

Indicates that the user ID of any user editing queue PARMLIB data for a job is a candidate for the MFUser field. This value would be the user ID of the last person to edit queue PARMLIB data for a job.

CA7

Indicates to select the user ID assigned to CA WA CA 7 Edition for the MFUser field.

APPL

(Optional) Identifies the CA WA CA 7 Edition 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 authority of the user to access CA WA CA 7 Edition.

BYPSEC

(Optional) Specifies functions for which to bypass UID security when accessing jobs in the database or queues.

Important! We do *not* recommend the use of these options. If selected, serious security exposures sometimes result. Decide to use these options only after careful consideration of the possible consequences of bypassing the CA WA CA 7 Edition security interface.

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

1

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

2

Indicates that security access to jobs is not validated during forecast processing.

3

Indicates that security access to requirement successor jobs is not validated during job 'purge' delete processing (job definition panel). That is, assume that 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 whether the current user has update access to job B.

CCLASS

(Optional) Specifies the resource class being used for security calls that are made from CA WA CA 7 Edition to validate a user's authority to access its calendars. The default resource class is CALENDAR for CA ACF2 and RACF users and $CALNDR for CA Top Secret users. This field can be up to eight characters.

DFLTUSER

(Optional) Specifies the default USERID. This USERID is used when it is requested in the SUBUID hierarchy. This field can be up to eight characters.

DISPLAY

(Optional) Determines whether the USERID is displayed when it is entered on the Logon panel.

YES

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

NO

Indicates that the USERID is not displayed.

EXTERNAL

(Optional) Identifies the security functions (calls) that external security is to control. Required for external security. CA WA CA 7 Edition internal security controls any security functions that are not specified on the external keyword. This control requires the presence of a CA WA CA 7 Edition security module built using the CA WA CA 7 Edition SECURITY macro. For more information, see the SECURITY macro.

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

AGENT

Indicates that any attempt to submit an agent job or execute an agent command is validated through external security.

CALENDAR

Indicates that any attempt to access a CA WA CA 7 Edition 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 value includes panel access throughout CA WA CA 7 Edition.

DATASET

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

LOGON

Indicates that logons to CA WA CA 7 Edition 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 WA CA 7 Edition LOGONs establishes the security environment for each CA WA CA 7 Edition user.

SUBCHECK

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

When a user requests a job through a DEMAND, LOAD, or RUN, CA WA CA 7 Edition attempts to determine the USERID with which the job runs. External security is used to validate the authority of the requester 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 examined. This verification prevents unauthorized usage of USERIDs.

If this option is used, CA WA CA 7 Edition also validates attempts to add, change, or update the RESPONSE ID associated with an ARFSET.

SUBOWNER

Performs the 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.

HIDEGRP

(Optional) Overlays user security group values coded in JCL with @ characters whenever JCL is listed with one of the inquiry commands.

NO

Displays the values. This value is the default.

YES

Hides the following in inquiry output:

GROUP keyword value in JOB statements

For a list of the affected inquiry commands, see keyword HIDEUSER.

HIDEPW

(Optional) Overlays user security password values coded in JCL with @ characters whenever JCL is listed with one of the inquiry commands.

NO

Displays the values. This value is the default.

YES

Hides the following in inquiry output:

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

For a list of the affected inquiry commands, see keyword HIDEUSER.

HIDEUPD

(Optional) Suppresses the last updater on several of the listing commands.

Note: After you enable this keyword, perform an update on the element. When this element is updated, *SECURE* is placed in the updater field.

NO

Displays the values. This value is the default.

YES

Places *SECURE* in the updater field after the next update.

HIDEUSER

(Optional) Overlays user security ID values coded in JCL statements with @ characters whenever JCL is listed with one of the inquiry commands.

NO

Displays the values. This value is the default.

YES

Hides the following 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

LGVAR

LJCK

LJCL

LLIB

LPDS

LPRRN,LIST=JCL

LQ,LIST=JCL

LQUE,LIST=JCL

LRDY,LIST=JCL

LREQ,LIST=JCL

JCLUID

(Optional) Prevents submission of jobs whose JCL contains a USERID. This parameter is only applicable if a SUBUID hierarchy is also specified.

YES

Indicates that CA WA CA 7 Edition submits the job after validating that CA WA CA 7 Edition has authority to submit for the USERID in the JCL. This value is the default.

NO

Indicates that CA WA CA 7 Edition does 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

(Optional) Suppresses the submit check on the LOAD(H) command. This option only has meaning if the EXTERNAL options include SUBCHECK.

YES

Validates the LOAD(H) commands with SUBCHECK. This value is the default.

NO

Exempts the LOAD(H) commands from the SUBCHECK validation.

LOGOPID

(Optional) Specifies whether the transaction log records for /LOGON commands include operator ID. In all cases, password values are not logged.

YES

Indicates that transaction log records for /LOGON commands include operator ID. YES does not write the operator ID to type x‘72’ log records. This value is the default.

NO

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

ALL

Indicates that transaction log records for all commands include operator ID. ALL includes type x‘72’ log records.

MIXPW

(Optional) Specifies whether the logon password can validly contain lowercase characters.

NO

Translates passwords to uppercase. This value is the default.

YES

Does not translate passwords to uppercase, allowing the password to contain lowercase characters. Verify that your security interface (CA ACF2, CA Top Secret, or RACF) supports lowercase passwords.

MULTIJOB

(Optional) Indicates whether CA WA CA 7 Edition controls the presence of several JOB statements within a JCL member. One exception is the following: JOB statements found within in-stream DD DATA are not controlled and are submitted as is.

IGNORE

Indicates that CA WA CA 7 Edition does not test for the presence of multiple JOB statements within a job member when submitting the cards to the internal reader. This value is the default.

FLUSH

Indicates that CA WA CA 7 Edition submits only the first job within a JCL member and flush the rest of JCL. No special sign of JCL truncation is generated.

REQUEUE

Indicates that CA WA CA 7 Edition does not submit but requeues the job that has several JOB statements within a JCL member. The requeued job has R-MJOB status.

Note: The MULTIJOB=IGNORE option is sometimes desirable for sites that transmit jobs between MVS nodes.

PCLASS

(Optional) Specifies the resource class being used for security calls. The calls are made from CA WA CA 7 Edition to validate the authority of the user to access CA WA CA 7 Edition commands and panels. The default resource class is PANEL. This field can be up to eight characters.

Note: If you change this value, examine the RCLASS keyword. The default for the RCLASS keyword is PANEL.

PROPAGATE

(Optional) Pertains only to RACF (and other SAF environments). This value determines the method that CA WA CA 7 Edition uses 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 WA CA 7 Edition inserts a USER= parameter in the JOB statement when a job is submitted. This value is the default.

YES

Indicates that CA WA CA 7 Edition does not modify the JCL being submitted. Instead, the USERID is propagated to the submitted job because the USERID of the job is 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

(Optional) Determines whether a USERID is required to be the same as the OWNER to access a job on the CA WA CA 7 Edition/Personal Scheduling panel.

YES

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

NO

Indicates no validation.

RCLASS

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

RESLOGON

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

YES

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

NO

/LOGON command is permitted.

RLOGUID

(Optional) Determines whether the LRLOG command (List Run Log) subjects job-related events to CA WA CA 7 Edition UID internal security checks. For more information about the LRLOG command, see the *Command Reference Guide*.

YES

Performs UID checking for LRLOG. This value causes LRLOG to display only jobs that the LRLOG requestor has access to. This value is the default.

NO

Performs no UID checking for LRLOG.

SCLASS

(Optional) Specifies the resource class being used for security calls that are made from CA WA CA 7 Edition to validate a user's authority to submit CA WA CA 7 Edition jobs under other user IDs. The default resource class is SUBMIT. This field can be up to eight characters.

STATSID

(Optional) Controls disposition of USERID in PDS directory data when using the CA WA CA 7 Edition editor. Members that are built with CA Endevor in a CA Endevor library do not have the USERID placed in the STATS.

YES

Writes the USERID to the PDS directory. This value is the default.

NO

Does not write the USERID to the PDS directory but writes it out as all @s.

SUBNOID

(Optional) Specifies the disposition of jobs that 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 can 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 insert a USERID manually into the JCL from the Queue JCL Edit panel. CA WA CA 7 Edition identifies the USERID and satisfies the R-NOUID requirement. This value is the default.

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

YES

Indicates that jobs can be submitted without a USERID.

SUBUID

(Optional) Specifies a hierarchy of candidate USERID sources for USERID insertion during job submission. If CA WA CA 7 Edition is 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 select.

If the SUBUID keyword is added to the SECURITY statement and CA WA CA 7 Edition is recycled, any jobs already in the request queue are not affected. Cancel them and demand them back in to use the new security data.

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

OWNER

Indicates to select the job OWNER ID 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) that then triggers additional jobs. The triggered jobs inherit the USERID. For data set triggers, jobs that create or "post" a data set to CA WA CA 7 Edition have their associated USERID propagated to any later 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 value would be the USERID of the last person to edit Queue JCL for a job.

DEFAULT

Indicates that the default USERID specified with the DFLTUSER keyword is to be selected for insertion.

CA7

Indicates that the USERID assigned to CA WA CA 7 Edition can be selected for USERID insertion. If selected, the CA WA CA 7 Edition USERID is inserted into the job's JCL during job submission. For CA ACF2, if started task checking is activated, this option cannot be used.

UID

(Optional) Specifies a UID Resource Table that was built using the CA7RTBL macro. If this parameter is specified, the UID Resource Table is loaded during CA WA CA 7 Edition initialization.

The CAL2OPTN member AL2UM09 can be used to build a table. A sample table (SASSRTBL) is also supplied in the CAL2LOAD library.

USER

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

XBSCLASS

(Optional) Controls whether all CAICCI and TCP/IP terminal sessions use the external security class used for Submit checking by CA WA CA 7 Edition. The SCLASS keyword on the SECURITY statement sometimes overrides this class.

YES

Communicates the CA WA CA 7 Edition Submit security class to the CAICCI and TCP/IP terminal sessions. This value is the default.

NO

Prevents the communication of the Submit security class setting from CA WA CA 7 Edition to the CAICCI and TCP/IP terminal sessions. This value means that the sessions use the default class ‘SUBMIT’. (This method is the way processing occurred before Ver.)

XBSUBCHK

(Optional) Controls whether the Batch Submit Checking option (BSUBCHK) on the LPAR where CA WA CA 7 Edition executes controls the submit checking on all CAICCI and TCP/IP terminal sessions regardless of the LPAR on which they execute.

YES

Communicates the BSUBCHK setting from the LPAR where CA WA CA 7 Edition executes to the CAICCI and TCP terminal sessions. This value is the default.

NO

Prevents the communication of the BSUBCHK setting from CA 7 to the CAICCI and TCP/IP terminal sessions. This value means the BSUBCHK setting on the LPAR where the CAICCI or TCP/IP interface executes is used. (This method is the way processing occurred before Version 12.0.)

XPSSID

(Optional) Defines the one- to eight-character USERID to use in the terminal logon for XPS job submission when no USERID is supplied on the submission request from the XPS CLIENT (typically CA NSM JMO or CA Workload Automation AE). This value is regarded as the requester ID for purposes of USERID insertion and propagation. The default value is **NONE**, which means there is no default USERID. Any XPS submission requests without an explicit USERID are rejected.

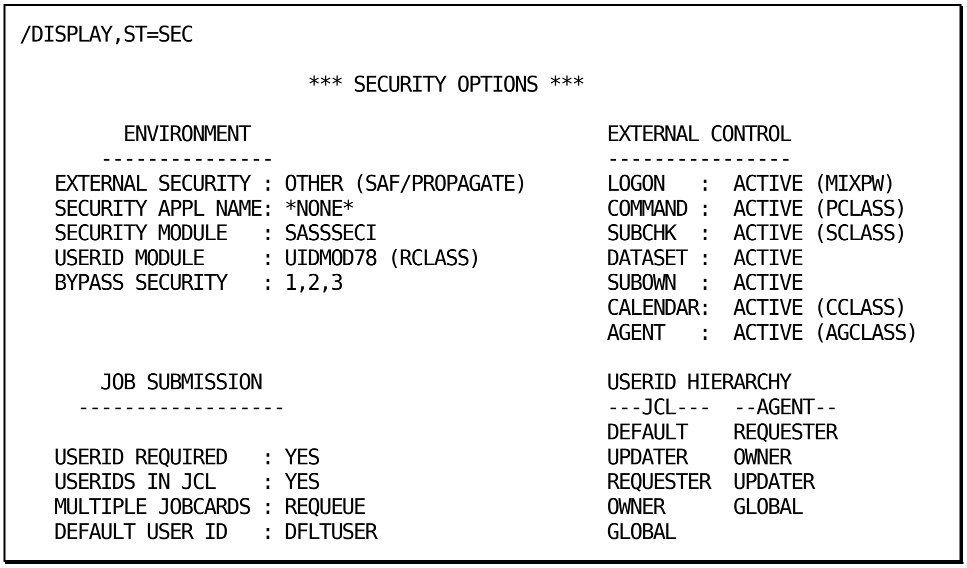
/DISPLAY,ST=SEC

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

This command has the following format:

```
/DISPLAY,ST={SEC|SECURITY}
```

The following is the /DISPLAY,ST=SEC panel.



```
/DISPLAY,ST=SEC

                         *** SECURITY OPTIONS ***

         ENVIRONMENT                               EXTERNAL CONTROL
       ---------------                             ----------------
   EXTERNAL SECURITY : OTHER (SAF/PROPAGATE)       LOGON   :  ACTIVE (MIXPW)
   SECURITY APPL NAME: *NONE*                      COMMAND :  ACTIVE (PCLASS)
   SECURITY MODULE   : SASSSECI                    SUBCHK  :  ACTIVE (SCLASS)
   USERID MODULE     : UIDMOD78 (RCLASS)           DATASET :  ACTIVE
   BYPASS SECURITY   : 1,2,3                       SUBOWN  :  ACTIVE
                                                   CALENDAR:  ACTIVE (CCLASS)
                                                   AGENT   :  ACTIVE (AGCLASS)

       JOB SUBMISSION                              USERID HIERARCHY
     ------------------                            ---JCL---  --AGENT--
                                                   DEFAULT    REQUESTER
   USERID REQUIRED   : YES                         UPDATER    OWNER
   USERIDS IN JCL    : YES                         REQUESTER  UPDATER
   MULTIPLE JOBCARDS : REQUEUE                     OWNER      GLOBAL
   DEFAULT USER ID   : DFLTUSER                    GLOBAL
```

This panel contains the following fields:

ENVIRONMENT

Lists the options indicating the external security package present, if any, and the security application name for CA WA CA 7 Edition defined to external security.

EXTERNAL SECURITY

Indicates the external security package that is found during CA WA CA 7 Edition initialization. The following are the possible values:

CA Top Secret

Indicates CA Top Secret security.

CA ACF2

Indicates CA ACF2 security.

OTHER (SAF)

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

OTHER (SAF/PROPAGATE)

Indicates PROPAGATE=YES is included on the SECURITY statement.

NONE

Indicates no external security.

SECURITY APPL NAME

Identifies the application name for CA WA CA 7 Edition, defined to the external security package.

SECURITY MODULE

Identifies the security module built using the CA WA CA 7 Edition SECURITY macro and identified by the NAME keyword on the SECURITY statement in the initialization file.

USERID MODULE

Identifies the USERID module built using the CA WA CA 7 Edition USERID macro that defines any correspondence between various UID values under CA WA CA 7 Edition internal security. The resource (class) name under which security calls are made is displayed as (RCLASS). This information is stated as the RCLASS keyword on the SECURITY statement in the initialization file.

If you use a RACF system, the third character is translated to an @ (at sign) when the actual call is made.

BYPASS SECURITY

Indicates that options are coded in the BYPSEC keyword on the SECURITY statement in the initialization file. We do not recommend these options, and the label is not present when you selected no options.

1

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

2

Indicates that security access to jobs is not validated during forecast processing.

3

Indicates that security access to requirement successor jobs is not validated during job "purge" delete processing.

EXTERNAL CONTROL

Identifies the status of external security control for specific security functions under CA WA CA 7 Edition. The status is ACTIVE for each function that external security controls and INACTIVE for those functions that CA WA CA 7 Edition internal security controls.

LOGON

Indicates login and signoff.

If (MIXPW) appears after LOGON: ACTIVE, CA WA CA 7 Edition honors mixed case passwords when logging in to CA WA CA 7 Edition.

COMMAND

Indicates command and panel security.

If command/panel security is active, the current PCLASS name from the SECURITY statement follows in parentheses.

If you use a RACF system, the third character is translated to an @ (at sign) when the actual call is made.

SUBCHK

Indicates job submission authority.

If job submission security is active, the current SCLASS name from the SECURITY statement follows in parentheses.

If you use a RACF system, the third character is translated to an @ (at sign) when the actual call is made.

DATASET

Indicates data set security.

SUBOWN

Indicates job owner ID security.

CALENDAR

Indicates CA WA CA 7 Edition base calendar security.

If calendar security is active, the current CCLASS name from the SECURITY statement follows in parentheses.

If you use a RACF system, the third character is translated to an @ (at sign) when the actual call is made.

AGENT

Indicates CA WA CA 7 Edition base agent security.

If agent security is active, the current AGCLASS name from the SECURITY statement follows in parentheses.

JOB SUBMISSION

Indicates whether a USERID is required for a job submission.

USERID REQUIRED

Indicates whether a USERID is required in the job.

YES

Requires a USERID. Jobs are not submitted without a USERID.

NO

Does not require a USERID. Jobs are submitted without a USERID.

N/A

Indicates that JCL USERID hierarchy is set to null.

USERIDS IN JCL

Indicates whether a USERID is coded in JCL.

YES

Submits jobs if a USERID is coded in JCL.

NO

Does not submit jobs when a USERID is coded in JCL.

N/A

Indicates that JCL USERID HIERARCHY is set to null.

MULTIPLE JOBCARDS

Indicates whether CA WA CA 7 Edition controls presence of multiple JOB statements within JCL.

IGNORE

Indicates CA WA CA 7 Edition does not control jobs at submission time with respect to JOB statements.

FLUSH

Indicates that CA WA CA 7 Edition submits only the first job within its JCL. The rest of JCL is truncated.

REQUEUE

Indicates that CA WA CA 7 Edition does not submit a job having several JOB statements within its JCL, but instead the job is requeued with R-MJOB status.

DEFAULT USER ID

Identifies the default USERID defined with the DFLTUSER keyword on the SECURITY statement.

USERID HIERARCHY

Lists up to five sources in the left column, in the priority order, from which to select a USERID for JCL insertion during job submission. The SUBUID keyword on the SECURITY statement in the initialization file sets this priority.

If agent security is active, the column on the right shows the selection priority for Mainframe Userid for Agent jobs. The AGUSER keyword on the SECURITY statement in the initialization file sets this priority.

An entry of *NONE* in either column indicates that the USERID hierarchy for that column is inactive or set to null.

DEFAULT

Specifies the default USERID defined with the DFLTUSER keyword on the SECURITY statement.

UPDATER

Specifies the USERID of the last queue JCL updater.

REQUESTER

Specifies the USERID of a user requesting a job or the USERID from a triggering job. Indicates that CA WA CA 7 Edition submits only the first job within its JCL. The rest of JCL is truncated.

OWNER

Specifies the job owner USERID.

GLOBAL

Specifies the CA WA CA 7 Edition USERID.

Chapter 4: Implementing Security with CA Top Secret

This chapter describes the steps necessary to implement the CA WA CA 7 Edition 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.

This section contains the following topics:

Define CA 7 to CA Top Secret

The following topics provide examples of CA Top Secret commands that can be used to implement CA WA CA 7 Edition external security.

Note: For more information about the commands listed in this chapter, see the *CA Top Secret Command Function Guide* (for z/OS).

The security definitions provided in this section are recommendations for establishing your CA WA CA 7 Edition security environment. Each site is responsible for determining whether these recommendations meet local auditing and security standards.

Define the CA WA CA 7 Edition Facility

The CA Top Secret Facility Matrix lets installations to define CA WA CA 7 Edition as a special facility for CA Top Secret to protect. The facility definition is used to define specific execution requirements for CA WA CA 7 Edition such as authorization to submit jobs, identify CA WA CA 7 Edition 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 WA CA 7 Edition facility:

Note: The definition of the CA WA CA 7 Edition facility using the TSS modify command is not permanent. We recommend that you add the CA WA CA 7 Edition facility definition commands to the CA Top Secret startup parameter file to verify that the CA WA CA 7 Edition facility is defined during CA Top Secret initialization. For more information about the TSSPARM0 parameter file, see *CA Top Secret Getting Started*.

This command has the following format:

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

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 WA CA 7 Edition facility.

ASUBM

Identifies the CA WA CA 7 Edition facility as authorized for Job Submission.

MULTIUSER

Identifies the CA WA CA 7 Edition facility as a multiuser address space.

NOABEND

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

PGM=SAS

Identifies the first three characters of the program name that issues SVC calls for security validations under the CA WA CA 7 Edition facility. Required.

LOG(ALL)

Specifies that CA Top Secret logs all security events for the CA WA CA 7 Edition facility.

Define the CA WA CA 7 Edition ACID

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

This command has the following format:

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

TSS CREATE

Indicates the CA Top Secret command used to create ACIDs.

CA7ONL

Indicates the name chosen in this example for the CA WA CA 7 Edition online ACID.

NAME

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

FAC(STC,BATCH)

Specifies the CA Top Secret command parameter used to define the CA WA CA 7 Edition ACID as a started task and to allow batch job submission.

TYPE(USER)

Identifies the CA WA CA 7 Edition ID as a User ACID.

PASS(NOPW)

Indicates that the CA WA CA 7 Edition ACID does not require a password.

DEPT(CA7OPS)

Establishes the owning department for the CA WA CA 7 Edition ACID.

MASTFAC(CA7)

Identifies the "master facility" for the CA WA CA 7 Edition online ACID. This value is the facility ID defined previously.

NOSUBCHK

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

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

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

TSS ADDTO(STC)

Specifies the CA Top Secret command used to add information to the Started Task Facility.

PROCNAME(CA7ONL)

Identifies the procedure name to use for the CA WA CA 7 Edition started task.

ACID(CA7ONL)

Associates the started task procedure name with the CA WA CA 7 Edition ACID.

Define ICOM to CA Top Secret

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

This command has the following format:

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

TSS CREATE

Specifies the CA Top Secret command used to create ACIDs.

CA7ICOM

Specifies the name chosen in this example for the CA WA CA 7 Edition ICOM ACID.

NAME

Specifies the CA Top Secret Create command parameter used to describe the ACID you are creating. In this case, the 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 ICOM ID as a User ACID.

PASS(NOPW)

Indicates that the ICOM ACID does not require a password.

DEPT(CA7OPS)

Establishes the owning department for the ICOM ACID.

MASTFAC(CA7)

Identifies the master facility for the ICOM ACID. This value is the facility ID defined previously.

NOSUBCHK

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

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

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

TSS ADDTO(STC)

Specifies the CA Top Secret command used to add information to the Started Task Facility.

PROCNAME(CA7ICOM)

Identifies the procedure name to be used for the ICOM started task.

ACID(CA7ICOM)

Associates the started task procedure name with the ICOM ACID.

Control User Access to CA WA CA 7 Edition

Because CA WA CA 7 Edition 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 CA Standard Security Facility (SSF), and a CA WA CA 7 Edition initialization parameter. For more information about the CA WA CA 7 Edition initialization parameters, see the security initialization options chapter.

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

This command has the following format:

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

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 gain access to the CA WA CA 7 Edition facility.

FAC

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

(CA7)

Specifies the name used in this example for the CA WA CA 7 Edition facility as defined previously.

Define CA 7 Command Security

CA WA CA 7 Edition command security includes security for top line commands, panel access, and functions within a panel. All commands have a unique resource name that 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 WA CA 7 Edition. Panels have a unique panel-ID that can be specified under CA Top Secret as a resource name to restrict access to CA WA CA 7 Edition applications. Permitting access to a resource does not grant full functional authority for a given command or panel. Each panel can 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.

For a list of CA WA CA 7 Edition commands and panels and their associated resource names, see Security Tables (see page 153).

The following examples illustrate the use of the CA Top Secret PERMIT command to authorize access to CA WA CA 7 Edition commands and panels.

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

This command has the following format:

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

TSS PERMIT

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

CA7USER

Specifies the user ACID to receive 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 WA CA 7 Edition. If you have specified a resource type other than PANEL (see the SECURITY statement PCLASS keyword), substitute that value for PANEL.

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

This command has the following format:

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

TSS PERMIT

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

CA7USER

Specifies the user ACID to receive 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 WA CA 7 Edition.

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

Lets you specify certain controlled accesses such as a time of day. For more information about the CONTROL parameter, see the *CA Top Secret Command Options Guide (for z/OS)*.

Define the CA WA CA 7 Edition Agent Job Submission/Command Security

If agent job submissions and command executions are being validated, authorizations are performed to verify that the mainframe user (MFUser) is authorized to submit agent jobs to the specific agent name using the agent user ID. Authorizations are also performed to verify that the signed on user is authorized to perform any agent command executions. The AGCLASS keyword on the SECURITY statement determines the resource class used for these authorizations.

The following are examples of CA WA CA 7 Edition agent job submission and agent command execution rules:

This example illustrates defining ownership and then giving job submission authority using a specific agent user ID and specific agent name:

```
TSS ADDTO(CA7DEPT) AGENT(CA71.AGENTUSR)
```

TSS ADDTO

Specifies the CA Top Secret command used to define ownership for a resource.

CA7DEPT

Specifies the CA Top Secret ACID to receive ownership for a resource.

AGENT(CA71.AGENTUSR)

Specifies the resource class AGENT followed by the resource name.

```
TSS PERMIT(CA7USER) AGENT(CA71.AGENTUSR.AGTUSER1.UNIXAGT) ACCESS(READ)
```

TSS PERMIT

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

CA7USER

Specifies the user ACID to receive access to submit the agent job.

AGENT(CA71.AGENTUSR.AGTUSER1.UNIXAGT)

Specifies the resource class AGENT followed by the resource name in the following format:

ca7-instance-id.AGENTUSR.*agent-userid*.*agent-name*

ACCESS(READ)

Specifies the access level. READ is required for agent job submission.

This example illustrates defining ownership and giving agent command execution authority for a specific agent name:

```
TSS ADDTO(CA7DEPT) AGENT(CA71.AGENTMSG)
```

TSS ADDTO

Specifies the CA Top Secret command used to define ownership for a resource.

CA7DEPT

Specifies the CA Top Secret ACID to receive ownership for a resource.

AGENT(CA71.AGENTMSG)

Specifies the resource class AGENT followed by the resource name.

```
TSS PERMIT(CA7USER) AGENT(CA71.AGENTMSG.CONTROLSHUTDOWN.UNIXAGT) ACCESS(READ)
```

TSS PERMIT

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

CA7USER

Specifies the user ACID to receive access to execute the agent command.

AGENT(CA71.AGENTMSG.CONTROLSHUTDOWN.UNIXAGT)

Specifies the resource class AGENT followed by the resource name in the following format:

ca7-instance-id.AGENTMSG.*verbsubverb.agent-name*

ACCESS(READ)

Specifies the access level. READ is required for agent command execution.

Secure the /MVS Command

The /MVS command allows a CA WA CA 7 Edition user to issue an MVS console command from a CA WA CA 7 Edition terminal. Although such a facility can prove indispensable in certain situations, the risks associated with an indiscriminate use of the command are obvious. This section discusses security concerns regarding the use of the command.

Note: For more information about the /MVS command, see the *Command Reference Guide*.

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

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

We recommend that you employ CA WA CA 7 Edition command security to restrict /MVS command access to a limited class of privileged users.

Calendar Security

To protect access to the CA WA CA 7 Edition calendars, use the CALENDAR option in the EXTERNAL keyword values on the SECURITY statement in the CA WA CA 7 Edition initialization file.

Set up the resource name of $CALNDR. The only service level checked is READ. If you have specified a resource type other than $CALNDR, substitute that value for $CALNDR.

The command to add this resource has the following format (substitute specified value for $CALNDR):

```
TSS ADD(RDT) RESCLASS($CALNDR) ATTR(MASK) +
ACLST(NONE,UPDATE,READ,WRITE,CONTROL,CREATE,SCRTCH,ALL) DEFACC(NONE)
TSS REPLACE(RDT) RESCLASS($CALNDR) ATTR(DEFPROT)
```

This example sets up a rule to access a calendar:

```
TSS PERMIT(ca7user) $CALNDR(calendar-name) ACCESS(READ)
```

Note: In previous releases, the resource name was CALENDAR. Because CA Top Secret now considers CALENDAR a reserved word, CA WA CA 7 Edition uses the $CALNDR resource name.

Control Job Submission Under CA WA CA 7 Edition

Depending on your security options, submit checking can be done. For these options, see the SECURITY statement.

This command has the following format:

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

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 receive 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.

If you have specified a resource type other than SUBMIT for SCLASS security checking, the checking is done as a generic resource check.

The command to add this resource has the following format (substitute specified value for SUBMIT):

```
TSS ADD(RDT) RESCLASS(SUBMIT) ATTR(MASK)+
ACLST(NONE,UPDATE,READ,WRITE,CONTROL,CREATE,SCRTCH,ALL) DEFACC(NONE)
TSS REPLACE(RDT) RESCLASS(SUBMIT) ATTR(DEFPROT)
```

The command to permit a user to use this resource has the following format (substitute specified value for SUBMIT):

```
TSS PERMIT(USERID1) SUBMIT(USERID2) ACCESS(READ)
```

Program Protection

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

This command has the following format:

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

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 WA CA 7 Edition online ACID.

PROG

Specifies the CA Top Secret keyword that identifies the resource to which this command applies.

UCC7

Identifies the CA WA CA 7 Edition main driver module.

This command has the following format:

```
TSS PERMIT(CA7ONL) PROG(SASS)
```

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 WA CA 7 Edition online ACID.

PROG(SASS)

Identifies a program prefix of SASS. This command grants CA WA CA 7 Edition the authority to execute any program with a prefix of SASS.

Batch Users

Batch USERIDs that are associated with jobs submitted through CA WA CA 7 Edition require access to the program SASSJJCL. SASSJJCL is the LOAD program that identifies resources used by a job to CA WA CA 7 Edition.

To prevent the unauthorized use of CA WA CA 7 Edition, we recommend that you secure access to the following programs:

- SASSBCLP - Batch Card Load Program
- SASSBSTR - Batch Terminal Interface
- SASSTRLR - Trailer Step Facility
- U7SVC - CA WA CA 7 Edition SVC Facility
- CAL2X2W0 - CA WA CA 7 Edition CAICCI Interface
- CAL2X2T0 – CA WA CA 7 Edition TCP/IP Interface

Job Submission

CA WA CA 7 Edition maintains a record of USERIDs that can be associated with a CPU job from queue entry to job submission. This association is not applicable to XPJOB or agent job definitions. XPJOB job and agent job submission security is described in the topic Security Considerations for Cross-Platform Scheduling (see page 18). For agent jobs, also see Define the Agent Submission/Command Security (see page 53).

If requested, a USERID can be inserted into the JCL of a job, before submission, to satisfy batch security requirements on your system. CA WA CA 7 Edition has five potential sources for USERIDs:

Job Owner

Specifies a USERID from the OWNER field for a job on the job definition panel. (SUBUID value of OWNER)

JCL ID

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

Requester

Specifies the USERID of the user that requests a job through the DEMAND, LOAD, or RUN commands. (SUBUID value of REQ)

Queue JCL

Specifies the USERID of a user editing the queued JCL of a job in the CA WA CA 7 Edition request queue. (SUBUID value of QJCL)

CA-7

Specifies the USERID assigned to CA WA CA 7 Edition at startup. If requested, the CA WA CA 7 Edition ID is propagated to submitted jobs. (SUBUID value of CA7)

The specification of the SUBUID keyword on the SECURITY statement in the initialization file determines the priority of USERID sources. The SUBUID keyword specifies a hierarchy of USERIDs for JCL insertion.

At submission time, CA WA CA 7 Edition 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 JCL of a job, and the job is submitted, assuming all other requirements are 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 WA CA 7 Edition checks the status of the SUBNOID flag. The SUBNOID keyword specified on the SECURITY statement in the initialization file sets the SUBNOID flag. If SUBNOID=YES, a job can be submitted without a USERID. If SUBNOID=NO, jobs cannot be submitted without a valid USERID. The jobs 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.

Satisfy the R-NOUID Requirement

Two methods can satisfy the R-NOUID requirement.

- If you have specified the QJCL keyword on the SUBUID parameter, you can FETCH/EDIT the queued JCL of the job and immediately do a SAVE/REPLACE. CA WA CA 7 Edition saves the USERID of the user editing the queued JCL. This method 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 add a USERID to the queued JCL of the job manually. CA WA CA 7 Edition recognizes the addition of the USERID to the JCL. The R-NOUID requirement is satisfied.

Note: If the JCL of a job contains a USERID at queue entry time, this USERID overrides all other USERIDs. USERID insertion does not take place.

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

USERID Propagation

Establishing USERIDs to be associated with jobs when submitted by CA WA CA 7 Edition is a critical aspect of security. Defining the USERID hierarchy requires careful planning to verify 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 method 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 that triggers any additional jobs.
- For data set triggers, the USERID associated with a job that CA WA CA 7 Editionsubmitted and created the data set.
- For data set triggers that are initiated through U7SVC and SASSBCLP, the USERID associated with the user posting the creation of the data set.

JCL USERID Format

For USERID insertion, CA WA CA 7 Edition 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 a USER= statement follows to supply the USERID. The following example illustrates the JCL statement format used during ID insertion.

```
// USER=userid
```

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

UID Resources

UID security validation controls access to information about the CA WA CA 7 Edition database. 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 compared to the UID value associated with the job. This comparison provides job-level security for the CA WA CA 7 Edition database. External security does not provide an equivalent JOB level protection. For this reason, it is important that each CA WA CA 7 Edition user is assigned a UID that relates to the user's area of responsibility.

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

The UID value (0-999) can be obtained from the USERID entry in the internal security module, through UID resource validation during logons to CA WA CA 7 Edition or through the /UID top line command issued under a CA WA CA 7 Edition session. If you want to maintain USERIDs in the internal security module, see the internal security chapter for information about 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 that CA WA CA 7 Edition references during UID resource validation. This table contains resource names and associated UID value entries to use during the UID validation process. A sample resource table, SASSRTBL, is provided in both source and load module format and can 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 can change the resource class for calls to external security using the RCLASS= parameter on the SECURITY statement of the initialization file.

You can use the /PROF command to establish and maintain a default UID resource for users logging on to CA WA CA 7 Edition.

Note: For a description of the /PROF command, see the *Command Reference Guide*.

If you use COIDs, the CA WA CA 7 Edition USERID security module described in the USERID macro is required.

CA7RTBL Macro

The CA7RTBL macro is used to generate the UID Resource Table. For the required parameters of this macro, see the following descriptions. Once the new source has been created, see member AL2UM09 in the CA WA CA 7 Edition CAL2OPTN file for applying the USERMOD.

The following is an example of the CA7RTBL macro statement:

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

CA7RTBL

Specifies the UID Resource Table generation macro, which is used to build the UID Resource Table.

RSRC

Defines the resource name to generate in this entry of the table. The resource name can be a one- to eight-character name that meets site-specific external security resource naming conventions.

Resource names must not conflict with existing panel or command names when the default PANEL resource class is used.

Note: For more information, see the description of the RCLASS keyword in the SECURITY statement.

UID

Specifies the value that is associated with the resource name supplied on the RSRC= parameter. The value can be from 0 through 999. 255 is a special UID that grants all access.

Usage Notes

The UID Resource Table name must be a valid PDS member name.

The CA7RTBL macro must be coded starting in column 10.

Duplicate resource names are not permitted, but duplicate UID values are permitted in the table.

The UID Resource Table source must be assembled and link edited into a load library accessible by CA WA CA 7 Edition.

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.

The UID Resource Table name must be identified on the SECURITY statement of the initialization file using the UID= parameter.

The resource names coded in the UID Resource Table must be defined to external security.

UID Resource Table - SASSRTBL Source

The following is a sample 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=CA70255,UID=255                                   00136808
                    .                                                   00137008
         CA7RTBL RSRC=CA70299,UID=299                                   00142808
         CA7RTBL RSRC=CA70300,UID=300                                   00142908
         CA7RTBL RSRC=LAST                                              00143008
         END                                                            00150000
```

CA 7 Logon and UID Resource Validation

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

If the user is defined in the Internal Security module, any UID resource name passed on the Logon panel 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.

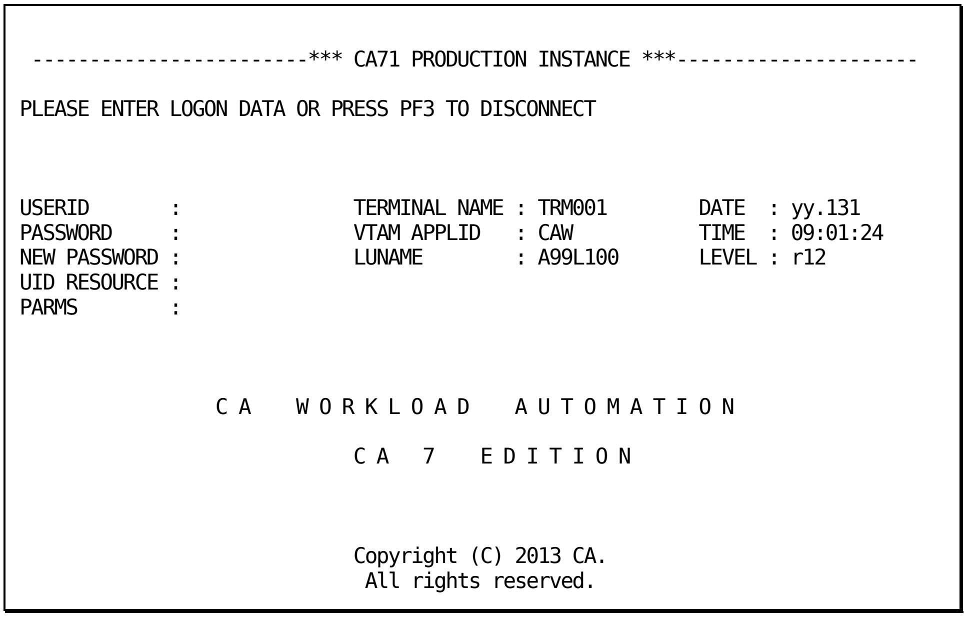
If the resource name is not found in the UID Resource Table, the user is signed on to CA WA CA 7 Edition with a UID value of 0. If a matching resource entry is found, a call is made to external security to validate the authority of the user to access the resource.

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 lets external security control UID assignment to CA WA CA 7 Edition users and eliminates the need to maintain all USERIDs in the CA WA CA 7 Edition Internal Security module.

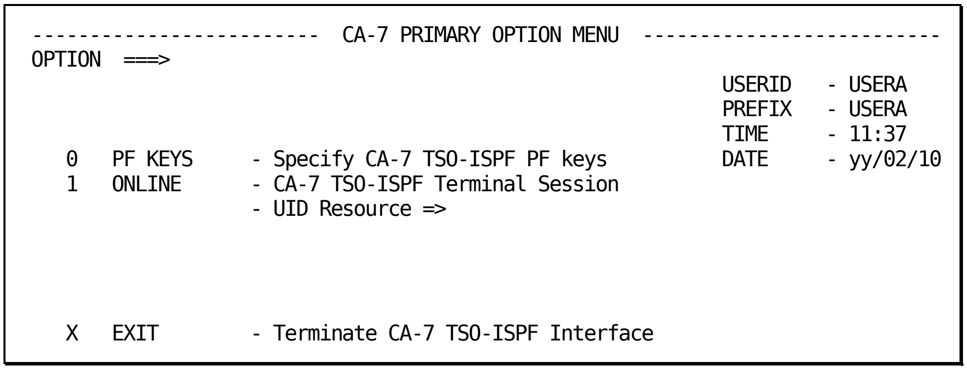
Note: The UID resource validation process is the same under the CA WA CA 7 Edition ISPF Interface; however, no password is required.

The following is a sample CA WA CA 7 Edition Logon panel:



```
 -----------------------*** CA71 PRODUCTION INSTANCE ***--------------------

PLEASE ENTER LOGON DATA OR PRESS PF3 TO DISCONNECT



USERID       :                TERMINAL NAME : TRM001        DATE  : yy.131
PASSWORD     :                VTAM APPLID   : CAW           TIME  : 09:01:24
NEW PASSWORD :                LUNAME        : A99L100       LEVEL : r12
UID RESOURCE :
PARMS        :




                 C A   W O R K L O A D   A U T O M A T I O N

                             C A   7   E D I T I O N




                             Copyright (C) 2013 CA.
                              All rights reserved.
```

The following is a sample CA WA CA 7 Edition ISPF Interface Primary Option Menu panel:



```
 -------------------------  CA-7 PRIMARY OPTION MENU  -------------------------
 OPTION  ===>
                                                            USERID   - USERA
                                                            PREFIX   - USERA
                                                            TIME     - 11:37
    0   PF KEYS      - Specify CA-7 TSO-ISPF PF keys        DATE     - yy/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 WA CA 7 Edition TSO-ISPF.

/UID Command

The /UID command can change the current UID processing value of a user through UID resource validation. The /UID command requires the following:

- The UID Resource Table option is implemented.
- The resources and the appropriate security authorization are defined to external security.

This command has the following format:

```
/UID,{R=resname|LIST}
```

R=*resname*

Identifies a resource name that exists in the UID Resource Table. The authorization of the user to access the resource is validated through external security. If authorized, the current UID value of the user is updated. The updated value reflects 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.

/REFRESH Command

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

This command has the following format:

```
REFRESH,MOD=xxxxxxxx
```

MOD=

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

xxxxxxxx

This name is the member name of the UID Resource Table. The member must reside in a load library accessible to CA WA CA 7 Edition.

External Communicators with CA Top Secret

The external communicators (SASSBSTR, SASSTRLR, U7SVC, CAL2X2W0, CAL2X2T0, and SASSBCLP) provide a means for users outside the CA WA CA 7 Edition address space to communicate with CA WA CA 7 Edition. Because the use of these programs sometimes permits access to production jobs, we recommend that you give careful consideration 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 can be implemented to prevent unauthorized use of these facilities. This section describes those controls.

Users of the CA WA CA 7 Edition CAICCI Interface require access to the program CAL2X2W0. This requirement is true regardless of whether this interface is invoked in batch (program CAL2X2WB), REXX (program CAL2X2WR), or program-to-program (CAL2X2WP).

Users of the CA WA CA 7 Edition TCP/IP interface require access to the program CAL2X2T0. This requirement is true regardless of whether this interface is invoked in batch (program CAL2X2TB), REXX (program CAL2X2TR), or program-to-program (CAL2X2TP).

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

- Terminal communication (Batch Terminal Interface, Trailer, CA WA CA 7 Edition CAICCI Interface, CA WA CA 7 Edition TCP/IP Interface, and U7SVC)
- Data set posting (U7SVC and SASSBCLP)

Terminal Communication

Each of the following lets you send terminal commands to CA WA CA 7 Edition:

- Batch Terminal Interface (SASSBSTR)
- Trailer facility (SASSTRLR)
- CAICCI Interface (CAL2X2W0)
- TCP/IP Interface (CAL2X2T0)
- U7SVC

Although no online terminal is used with this mode of communication, input from these programs is treated as terminal input by CA WA CA 7 Edition. Command security in these environments is handled as it is for all CA WA CA 7 Edition terminals. CA Top Secret controls access to CA WA CA 7 Edition commands when EXTERNAL=COMMAND is specified on the SECURITY statement in the initialization file. CA Top Secret determines a user's access to CA WA CA 7 Edition terminal commands based on the USERID supplied on the /LOGON command. Thus, when using an External Communicator, any command input must precede a /LOGON command.

CA Top Secret typically requires a password at logon. But including passwords in command input for the External Communicators would obviously represent a serious security exposure. Several checks are 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, a /LOGON statement is built using the USERID associated with the current user. Under certain conditions, it is not always 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 can 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 can 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 the control job submission under CA WA CA 7 Edition topic. If a /LOGON statement was generated or if the user's authority to use a USERID was successfully validated, CA WA CA 7 Edition allows the user to LOGON without a password.

The USERID of the current user is determined by using CAS9 SSF services.

Note: For more information about SSF, see the CA Common Services documentation.

Submit checking for External Communicators is controlled by the value of BSUBCHK that is set by CAIRIM.

Note: For more information, see the chapter "Execution" in the *Systems Programming Guide*.

SASSTRLR and External Security

The following information shows the actions that SASSTRLR performs 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 whether 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If so, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value of BSUBCHK for the instance is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y, and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

SASSBSTR, CAL2X2W0, CAL2X2T0, and External Security

The following information shows the actions that SASSBSTR, CAL2X2W0, and CAL2X2T0 perform during execution with relation to security.

A security EXTRACT is done to determine the USERID of the submitted job or user environment. 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 whether a logon statement was supplied. If no logon statement was supplied, 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If so, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value of BSUBCHK for the instance is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y, and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

For CAL2X2W0 (CAICCI Terminal) and CAL2X2T0 (TCP/IP Terminal) executions, the BSUBCHK setting for the target instance of CA WA CA 7 Edition can optionally be used.

If used, the setting overrides any setting on the submitting terminal. The submitting terminal BSUBCHK setting is used for either of the following:

- This option is disabled.
- In cases where either the sending terminal or the target CA WA CA 7 Edition instance are versions before Version 12.0.

If the target CA WA CA 7 Edition instance is using a value for submit class other than the default SUBMIT, that value can optionally be used for the submit check on the sending terminal. The default submit class value is used for either of the following:

- This option is disabled.
- Either the sending terminal or the target CA WA CA 7 Edition instance are versions before Version 12.0.

More information:

SECURITY Statement (see page 26)

U7SVC and External Security

The following information is intended to show the actions that U7SVC performs during execution with relation to security. Two different paths can be taken. The path depends on whether there is an input stream with the U7SVC or if it is only a D= to post a data set.

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 value of SVDSNCHK for the instance not Y, the D= command is passed through to CA WA CA 7 Edition with U7SVC doing no further security checking.

If the value of SVDSNCHK is Y, U7SVC makes a security call. This call determines whether 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 WA CA 7 Edition for processing.

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 whether 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If it is, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value BSUBCHK is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

Data Set Posting

The Batch Card Load Program (SASSBCLP) and U7SVC allow the user to post the creation of a data set to CA WA CA 7 Edition. Because such posting can satisfy requirements or cause job triggering, the need to secure the use of these facilities is critical. Two features of these facilities require a mention in this connection:

- Data set access validation.
- USERID propagation.

The USERID associated with the user of U7SVC or SASSBCLP is extracted to determine the authority of the user to create the data set. Under certain conditions, it is not always 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 initialization file, the USERID associated with the data set creation can be propagated to triggered jobs.

For example, suppose that a user whose USERID is XXX submits a batch job. The batch job uses U7SVC to post the creation of data set A.B to CA WA CA 7 Edition. 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 can be made to verify the authority of the user to create that data set whose creation is for posting to CA WA CA 7 Edition. For more information about the CAIRIM keywords that control this user ID verification, see the chapter "Execution" in the *Systems Programming Guide*.

Sample Definitions

The member AL2TSSS in the CA WA CA 7 Edition options file (CAL2OPTN) on the installation media contains sample TSS commands. The sample commands can be used to secure the CA WA CA 7 Edition processing environment under CA Top Secret. The definitions are intended as examples only. Review and modify them to meet the security requirements at your location. Once tailored to the specifications of your location, use the definitions as batch input to CA Top Secret.

Note: For more information about executing CA Top Secret commands in batch, see the *CA Top Secret Command Function Guide (for z/OS)*.

Chapter 5: Implementing Security with CA ACF2

This chapter describes the steps necessary to implement the CA WA CA 7 Edition 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 WA CA 7 Edition security environment. Each site is responsible for determining whether these recommendations meet local auditing and security standards.

This section contains the following topics:

Define CA 7 to CA ACF2

CA WA CA 7 Edition 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 that invokes CA WA CA 7 Edition. Multiple users can be logged on to CA WA CA 7 Edition at the same time. For this reason, it is referred to as a MUSASS - Multiple User Single Address Space Subsystem and requires the MUSASS attribute when defining the LOGONID.

CA WA CA 7 Edition also requires the authorization to submit jobs. This authorization can be accomplished by adding the JOBFROM permission to the CA WA CA 7 Edition LOGONID. This exempts CA WA CA 7 Edition Submit Validation and provides uninterrupted production scheduling. You can use the following CA ACF2 command example to define the CA WA CA 7 Edition LOGONID to CA ACF2.

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

The following is an example CA ACF2 command used to define the CA WA CA 7 Edition LOGONID:

```
INSERT  CA7  NAME(CA 7 Production ID)  STC  MUSASS
DUMPAUTH  JOBFROM
```

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 WA CA 7 Edition LOGONID.

NAME

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

STC

Identifies the CA WA CA 7 Edition LOGONID as a started task ID.

MUSASS

Defines the CA WA CA 7 Edition LOGONID as a Multiple User Single Address Space Subsystem.

DUMPAUTH

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

JOBFROM

Lets CA WA CA 7 Edition use the JOBFROM statement for USERID insertion at job submission time.

Note: If you already have a started task LOGONID with similar permissions and attributes, you can use the INSERT USING command option when defining the CA WA CA 7 Edition LOGONID. For a complete description of the USING command option and the additional privileges and attributes shown in this example, see the *CA ACF2 Administrator Guide (for z/OS)*.

Define CA 7 As a Resource

You can define CA WA CA 7 Edition to CA ACF2 as a resource to control LOGON access. The resource definition for CA WA CA 7 Edition under CA ACF2 is not required; however, it does provide an additional level of security for restricting access to CA WA CA 7 Edition. If CA ACF2 is to control LOGON security for CA WA CA 7 Edition, a resource check is made during LOGON. The resource check determines whether the user is authorized to access CA WA CA 7 Edition.

Follow these steps:

1. Define a Resource Rule under CA ACF2 identifying CA WA CA 7 Edition as a resource. If you are using CA ACF2 6.0 or higher, 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 WA CA 7 Edition initialization file and specify the resource name.
4. Add the LOGON keyword to the EXTERNAL= parameter list on the SECURITY statement in the CA WA CA 7 Edition initialization file.

The following is an example CA ACF2 application resource rule for CA WA CA 7 Edition:

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

$KEY(CA7)

Specifies the CA ACF2 keyword used to name the resource to protect. 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 permit or prevent 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. For more information about compiling and storing rules, see the *CA ACF2 Administrator Guide (for z/OS)*.

Define CA 7 Command Security

Implementation of CA WA CA 7 Edition command security includes securing top line commands, application panel access and panel functions. Each CA WA CA 7 Edition panel has a unique panel-ID that can be defined as a resource to CA ACF2. For subsequent functions on each panel, which can 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 WA CA 7 Edition first checks the authority of the user 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 choice 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 choice 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 WA CA 7 Edition database. This choice 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 WA CA 7 Edition but for the additional functions on each panel. Each command requires a service level entry on the resource rule definition to perform that function. For a list of the CA WA CA 7 Edition panel-IDs, commands, and access level requirements, see Security Tables (see page 153).

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, enter asterisks for each level of access you want to specify generically for users.

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

Note: CA ACF2 requires the use of Resident Directories to use Resource Rule masking. Resident Directories are sometimes required for other CA ACF2 options. For more information about Resource Rule Resident Directories, see the *CA ACF2 Administrator Guide (for z/OS)*.

The following is an example CA WA CA 7 Edition panel resource rule:

```
$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.
```

$KEY(L2DB1)

Identifies the Database Maintenance - Job Definition panel. The L2 preceding the DB1 is the CA WA CA 7 Edition product code and is required.

TYPE(PAN)

Identifies the type of resource rule. If you have specified a resource type other than PANEL (see the SECURITY statement PCLASS keyword), substitute the CA ACF2 SAFDEF assigned to this resource type for PAN.

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 WA CA 7 Edition panel and command resource rules under CA ACF2 require a resource type of PAN.

The following is an example CA WA CA 7 Edition "masked" panel resource rule:

```
*
$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.
*
```

$KEY(L2DB**)**

Identifies any CA WA CA 7 Edition 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. Other CA ACF2 options sometimes require Resident Directories. For more information about Resource Rule Resident Directories, see the *CA ACF2 Administrator Guide (for z/OS)*.

Secure the /MVS Command

The /MVS command allows a CA WA CA 7 Edition user to issue an MVS console command from a CA WA CA 7 Edition terminal. Although such a facility can prove indispensable in certain situations, the risks associated with an indiscriminate use of the command are obvious. This section discusses security concerns regarding the use of the command.

Note: For more information about the /MVS command, see the *Command Reference Guide*.

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

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

We recommend that you employ CA WA CA 7 Edition command security to restrict /MVS command access to a limited class of privileged users.

Define ICOM to CA ACF2

The CA WA CA 7 Edition Independent Communications Manager (ICOM) must also be defined to CA ACF2. ICOM is responsible for handling SMF data for jobs submitted through CA WA CA 7 Edition, and therefore requires an ACID to execute in a CA ACF2 secured environment. The following command example can be used to define ICOM to CA ACF2.

This command has the following format:

```
INSERT  CA7ICOM  NAME(CA 7 ICOM)  STC
DUMPAUTH  JOBFROM
```

INSERT

Specifies the CA ACF2 command used to add a LOGONID to the CA ACF2 database.

CA7ICOM

Specifies the name chosen in this example for the ICOM LOGONID.

NAME

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

STC

Identifies the ICOM LOGONID as a started task ID.

DUMPAUTH

Permits the ICOM LOGONID to generate complete dumps of its address space in the event of an abend.

JOBFROM

Allows ICOM to use the JOBFROM statement for USERID insertion at job submission time.

Define SUBMIT Resource Rules

You can prevent unauthorized access to LOGONIDs under CA WA CA 7 Edition. 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 that 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 use under CA WA CA 7 Edition. If you have specified a resource type other than SUBMIT (see the SECURITY statement SCLASS keyword), substitute the CA ACF2 SAFDEF assigned to this resource type for SUB.

An example of CLASMAP follows:

```
CLASMAP.SUB RESOURCE(SUBMIT) RSRCTYPE(SUB)
```

The following is an example CA WA CA 7 Edition SUBMIT resource rule:

```
  $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.
 *
```

$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
  *
  *
```

$KEY(USERID2)

Identifies the USERID for which the local UID string has submit authority.

TYPE(SUB)

Identifies as submit authority. If you have specified a resource type other than SUBMIT (see the SECURITY statement SCLASS keyword), use the RSRCTYPE(…) defined on the CLASMAP definition.

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.

Define the CA WA CA 7 Edition Agent Job Submission/Command Security

If agent job submissions and command executions are being validated, authorizations are performed to verify that the mainframe user (MFUser) is authorized to submit agent jobs to the specific agent name using the agent user ID. Authorizations are also performed to verify that the signed-on user is authorized to perform any agent command executions. The AGCLASS keyword on the SECURITY statement determines the resource class used for these authorizations. If the default AGCLASS FACILITY is not used, set up a class map in the ACF2 Options. An example follows where AGCLASS is the Resource Name and AGT is the resource type:

```
CLASMAP.AGT RESOURCE(AGCLASS) RSRCTYPE(AGT)
```

The following are examples of CA WA CA 7 Edition agent job submission and agent command execution rules:

This example illustrates giving job submission authority using a specific agent user ID and specific agent name:

```
$KEY(CA71) TYPE(AGT)
 AGENTUSR.agent-userid.agent-name UID(CA7USER) SERVICE(READ) ALLOW
```

$KEY(CA71) AGENTUSR.AGTUSER1.UNIXAGT)

Identifies the resource name in the following format:

(ca7-instance-id) AGENTUSR.*agent-userid.agent-name*

TYPE(AGT)

Identifies the type of resource rule. If you have specified a resource type other than AGENT (see the SECURITY statement AGCLASS keyword), substitute the CA ACF2 SAFDEF assigned to this resource type for AGT.

UID

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

SERVICE(READ)

Identifies the access level required to permit use of the resource.

ALLOW

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

This example illustrates giving agent command execution authority for a specific agent name:

```
$KEY(CA71) TYPE(AGT)
 AGENTMSG.CONTROL SHUTDOWN.UNIXAGT) UID(CA7USER) SERVICE(READ) ALLOW
```

$KEY(CA71) AGENTMSG.CONTROL SHUTDOWN.UNIXAGT

Identifies the resource name in the following format:

(ca7-instance-id) AGENTMSG.*verb subverb.agent-name*

TYPE(AGT)

Identifies the type of resource rule. If you have specified a resource type other than AGT (see the SECURITY statement AGCLASS keyword), substitute the CA ACF2 SAFDEF assigned to this resource type for AGT.

UID

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

SERVICE(READ)

Identifies the access level required to permit use of the resource.

ALLOW

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

Calendar Security

To protect access to the CA WA CA 7 Edition calendars, use the CALENDAR option in the EXTERNAL keyword values on the SECURITY statement in the CA WA CA 7 Edition initialization file.

Set up the resource name of CALENDAR. The only service level checked is READ. If you have specified a resource type other than CALENDAR (see the SECURITY statement CCLASS keyword), substitute that value for CALENDAR. Also substitute the CA ACF2 SAFDEF assigned to this resource type for CAL.

This example uses CLASMAP:

```
CLASMAP.CAL RESOURCE(CALENDAR) RSRCTYPE(CAL)
```

This example sets up a rule to access a calendar:

```
$KEY(calendar-name or mask) TYPE(CAL)
 UID(uid-mask) ALLOW SERVICE(READ)
```

Job Submission

CA WA CA 7 Edition maintains a record of USERIDs that can be associated with a CPU job from queue entry to job submission. This association is not applicable to XPJOB or agent job definitions. XPJOB job and agent job submission security is described in the topic Security Considerations for Cross-Platform Scheduling (see page 18). For agent jobs, also see Define the Agent Submission/Command Security (see page 53).

If requested, a USERID can be inserted into the JCL of a job, before submission, to satisfy batch security requirements on your system. CA WA CA 7 Edition has five potential sources for USERIDs:

Job Owner

Specifies a USERID from the OWNER field for a job on the job definition panel. (SUBUID value of OWNER)

JCL ID

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

Requester

Specifies the USERID of the user that requests a job through the DEMAND, LOAD, or RUN commands. (SUBUID value of REQ)

Queue JCL

Specifies the USERID of a user editing the queued JCL of a job in the CA WA CA 7 Edition request queue. (SUBUID value of QJCL)

CA-7

Specifies the USERID assigned to CA WA CA 7 Edition at startup. If requested, the CA WA CA 7 Edition ID is propagated to submitted jobs. (SUBUID value of CA7)

The specification of the SUBUID keyword on the SECURITY statement in the initialization file determines the priority of USERID sources. The SUBUID keyword specifies a hierarchy of USERIDs for JCL insertion.

At submission time, CA WA CA 7 Edition 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 JCL of a job, and the job is submitted, assuming all other requirements are 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 WA CA 7 Edition checks the status of the SUBNOID flag. The SUBNOID keyword specified on the SECURITY statement in the initialization file sets the SUBNOID flag. If SUBNOID=YES, a job can be submitted without a USERID. If SUBNOID=NO, jobs cannot be submitted without a valid USERID. The jobs 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.

Satisfy the R-NOUID Requirement

Two methods can satisfy the R-NOUID requirement.

- If you have specified the QJCL keyword on the SUBUID parameter, you can FETCH/EDIT the queued JCL of the job and immediately do a SAVE/REPLACE. CA WA CA 7 Edition saves the USERID of the user editing the queued JCL. This method 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 add a USERID to the queued JCL of the job manually. CA WA CA 7 Edition recognizes the addition of the USERID to the JCL. The R-NOUID requirement is satisfied.

Note: If the JCL of a job contains a USERID at queue entry time, this USERID overrides all other USERIDs. USERID insertion does not take place.

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

USERID Propagation

Establishing USERIDs to be associated with jobs when submitted by CA WA CA 7 Edition is a critical aspect of security. Defining the USERID hierarchy requires careful planning to verify 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 method 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 that triggers any additional jobs.
- For data set triggers, the USERID associated with a job that CA WA CA 7 Editionsubmitted and created the data set.
- For data set triggers that are initiated through U7SVC and SASSBCLP, the USERID associated with the user posting the creation of the data set.

JCL USERID Format

For USERID insertion, CA WA CA 7 Edition inserts a JOBFROM statement or a LOGONID statement immediately following the JOB statement. The ACF2CARD keyword on the SECURITY statement can control the type of statement used. (JOBFROM is the default.) The following example illustrates the JOBFROM and LOGONID JCL statement formats used during ID insertion for CA ACF2.

```
//*JOBFROM userid
```

or

```
//*LOGONID userid
```

UID Resources

UID security validation controls access to information about the CA WA CA 7 Edition database. 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 compared to the UID value associated with the job. This comparison provides job-level security for the CA WA CA 7 Edition database. External security does not provide an equivalent JOB level protection. For this reason, it is important that each CA WA CA 7 Edition user is assigned a UID that relates to the user's area of responsibility.

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

The UID value (0-999) can be obtained from the USERID entry in the internal security module, through UID resource validation during logons to CA WA CA 7 Edition or through the /UID top line command issued under a CA WA CA 7 Edition session. If you want to maintain USERIDs in the internal security module, see the internal security chapter for information about 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 that CA WA CA 7 Edition references during UID resource validation. This table contains resource names and associated UID value entries to use during the UID validation process. A sample resource table, SASSRTBL, is provided in both source and load module format and can 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 can change the resource class for calls to external security using the RCLASS= parameter on the SECURITY statement of the initialization file.

You can use the /PROF command to establish and maintain a default UID resource for users logging on to CA WA CA 7 Edition.

Note: For a description of the /PROF command, see the *Command Reference Guide*.

If you use COIDs, the CA WA CA 7 Edition USERID security module described in the USERID macro is required.

CA7RTBL Macro

The CA7RTBL macro is used to generate the UID Resource Table. For the required parameters of this macro, see the following descriptions. Once the new source has been created, see member AL2UM09 in the CA WA CA 7 Edition CAL2OPTN file for applying the USERMOD.

The following is an example of the CA7RTBL macro statement:

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

CA7RTBL

Specifies the UID Resource Table generation macro, which is used to build the UID Resource Table.

RSRC

Defines the resource name to generate in this entry of the table. The resource name can be a one- to eight-character name that meets site-specific external security resource naming conventions.

Resource names must not conflict with existing panel or command names when the default PANEL resource class is used.

Note: For more information, see the description of the RCLASS keyword in the SECURITY statement.

UID

Specifies the value that is associated with the resource name supplied on the RSRC= parameter. The value can be from 0 through 999. 255 is a special UID that grants all access.

Usage Notes

The UID Resource Table name must be a valid PDS member name.

The CA7RTBL macro must be coded starting in column 10.

Duplicate resource names are not permitted, but duplicate UID values are permitted in the table.

The UID Resource Table source must be assembled and link edited into a load library accessible by CA WA CA 7 Edition.

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.

The UID Resource Table name must be identified on the SECURITY statement of the initialization file using the UID= parameter.

The resource names coded in the UID Resource Table must be defined to external security.

UID Resource Table - SASSRTBL Source

The following is a sample 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=CA70255,UID=255                                   00136808
                    .                                                   00137008
         CA7RTBL RSRC=CA70299,UID=299                                   00142808
         CA7RTBL RSRC=CA70300,UID=300                                   00142908
         CA7RTBL RSRC=LAST                                              00143008
         END                                                            00150000
```

CA 7 Logon and UID Resource Validation

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

If the user is defined in the Internal Security module, any UID resource name passed on the Logon panel 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.

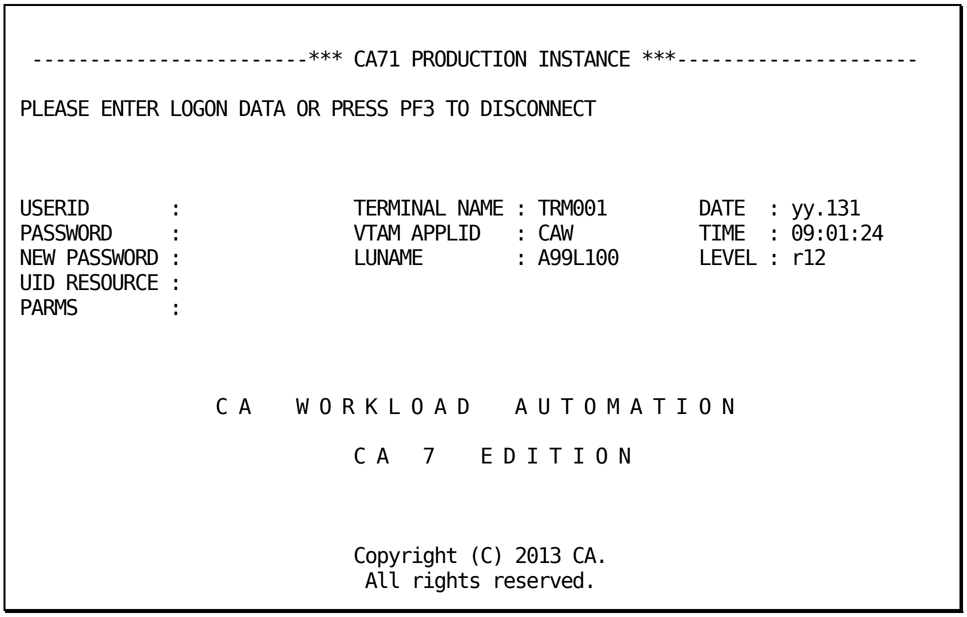
If the resource name is not found in the UID Resource Table, the user is signed on to CA WA CA 7 Edition with a UID value of 0. If a matching resource entry is found, a call is made to external security to validate the authority of the user to access the resource.

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 lets external security control UID assignment to CA WA CA 7 Edition users and eliminates the need to maintain all USERIDs in the CA WA CA 7 Edition Internal Security module.

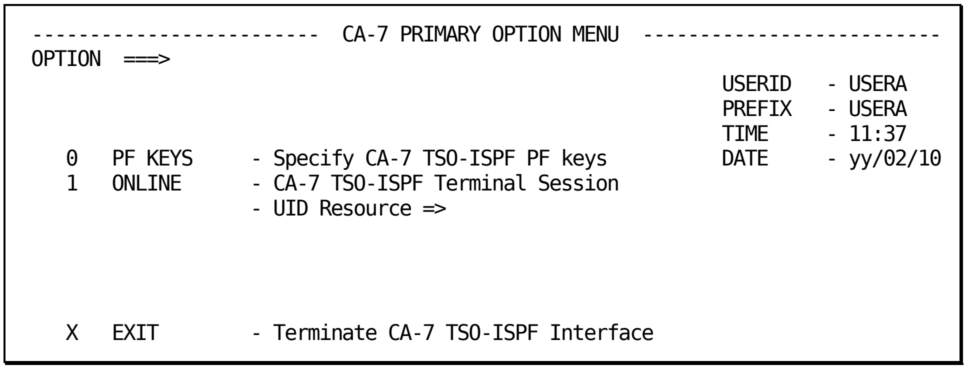
Note: The UID resource validation process is the same under the CA WA CA 7 Edition ISPF Interface; however, no password is required.

The following is a sample CA WA CA 7 Edition Logon panel:



```
 ------------------------*** CA71 PRODUCTION INSTANCE ***---------------------

PLEASE ENTER LOGON DATA OR PRESS PF3 TO DISCONNECT



USERID       :              TERMINAL NAME : TRM001       DATE  : yy.131
PASSWORD     :              VTAM APPLID   : CAW          TIME  : 09:01:24
NEW PASSWORD :              LUNAME        : A99L100      LEVEL : r12
UID RESOURCE :
PARMS        :



                 C A   W O R K L O A D   A U T O M A T I O N

                             C A   7   E D I T I O N



                             Copyright (C) 2013 CA.
                              All rights reserved.
```

The following is a sample CA WA CA 7 Edition ISPF Interface Primary Option Menu panel:



```
 ------------------------  CA-7 PRIMARY OPTION MENU  -------------------------
 OPTION  ===>
                                                            USERID   - USERA
                                                            PREFIX   - USERA
                                                            TIME     - 11:37
    0   PF KEYS     - Specify CA-7 TSO-ISPF PF keys         DATE     - yy/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 WA CA 7 Edition TSO-ISPF.

/UID Command

The /UID command can change the current UID processing value of a user through UID resource validation. The /UID command requires the following:

- The UID Resource Table option is implemented.
- The resources and the appropriate security authorization are defined to external security.

This command has the following format:

```
/UID,{R=resname|LIST}
```

R=*resname*

Identifies a resource name that exists in the UID Resource Table. The authorization of the user to access the resource is validated through external security. If authorized, the current UID value of the user is updated. The updated value reflects 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.

/REFRESH Command

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

This command has the following format:

```
REFRESH,MOD=xxxxxxxx
```

MOD=

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

xxxxxxxx

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

Program Protection

The following topics discuss program protection for CA WA CA 7 Edition and batch users.

CA 7 Requirements

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

Batch Users

Batch USERIDs that are associated with jobs submitted through CA WA CA 7 Edition require access to the program SASSJJCL. SASSJJCL is the LOAD program that identifies resources used by a job to CA WA CA 7 Edition.

To prevent the unauthorized use of CA WA CA 7 Edition, we recommend that you secure access to the following programs:

- SASSBCLP - Batch Card Load Program
- SASSBSTR - Batch Terminal Interface
- SASSTRLR - Trailer Step Facility
- U7SVC - CA WA CA 7 Edition SVC Facility
- CAL2X2W0 - CA WA CA 7 Edition CAICCI Interface
- CAL2X2T0 – CA WA CA 7 Edition TCP/IP Interface

External Communicators with CA ACF2

The external communicators (SASSBSTR, SASSTRLR, U7SVC, CAL2X2W0, CAL2X2T0, and SASSBCLP) provide a means for users outside the CA WA CA 7 Edition address space to communicate with CA WA CA 7 Edition. Because the use of these programs sometimes permits access to production jobs, we recommend that you give careful consideration 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 can be implemented to prevent unauthorized use of these facilities. This section describes those controls.

Users of the CA WA CA 7 Edition CAICCI Interface require access to the program CAL2X2W0. This requirement is true regardless of whether this interface is invoked in batch (program CAL2X2WB), REXX (program CAL2X2WR), or program-to-program (CAL2X2WP).

Users of the CA WA CA 7 Edition TCP/IP interface require access to the program CAL2X2T0. This requirement is true regardless of whether this interface is invoked in batch (program CAL2X2TB), REXX (program CAL2X2TR), or program-to-program (CAL2X2TP).

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

- Terminal communication (Batch Terminal Interface, Trailer, CA WA CA 7 Edition CAICCI Interface, CA WA CA 7 Edition TCP/IP Interface, and U7SVC)
- Data set posting (U7SVC and SASSBCLP)

Terminal Communication

Each of the following lets you send terminal commands to CA WA CA 7 Edition:

- Batch Terminal Interface (SASSBSTR)
- Trailer facility (SASSTRLR)
- CAICCI Interface (CAL2X2W0)
- TCP/IP Interface (CAL2X2T0)
- U7SVC

Although no online terminal is used with this mode of communication, input from these programs is treated as terminal input by CA WA CA 7 Edition. Command security in these environments is handled as it is for all CA WA CA 7 Edition terminals. CA ACF2 controls access to CA WA CA 7 Edition commands when EXTERNAL=COMMAND is specified on the SECURITY statement in the CA WA CA 7 Edition initialization file. CA ACF2 determines a user's access to CA WA CA 7 Edition terminal commands based on the LOGONID supplied on the /LOGON command. Thus, when using an External Communicator, a CA WA CA 7 Edition /LOGON command must precede any command input.

CA ACF2 typically requires a password at logon. But including passwords in command input for the External Communicators would obviously represent a serious security exposure. Several checks can 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, a /LOGON statement is built using the LOGONID associated with the current user. Under certain conditions, it is not 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, 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, it is assumed that the user has the authority to use the LOGONID. If the LOGONIDs differ, 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 WA CA 7 Edition allows the user to LOGON without a password.

The USERID of the current user is determined by using CAS9 CAISSF services.

Note: For more information about CAISSF, see the CA Common Services documentation.

The value of BSUBCHK that CAIRIM sets controls submit checking for External Communicators.

Note: For more information, see the chapter "Execution" in the *Systems Programming Guide*.

SASSTRLR and External Security

The following information shows the actions that SASSTRLR performs 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 whether 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If so, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value of BSUBCHK for the instance is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y, and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

SASSBSTR, CAL2X2W0, CAL2X2T0, and External Security

The following information shows the actions that SASSBSTR, CAL2X2W0, and CAL2X2T0 perform during execution with relation to security.

A security EXTRACT is done to determine the USERID of the submitted job or user environment. 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 whether a logon statement was supplied. If no logon statement was supplied, 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If so, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value of BSUBCHK for the instance is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y, and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

For CAL2X2W0 (CAICCI Terminal) and CAL2X2T0 (TCP/IP Terminal) executions, the BSUBCHK setting for the target instance of CA WA CA 7 Edition can optionally be used.

If used, the setting overrides any setting on the submitting terminal. The submitting terminal BSUBCHK setting is used for either of the following:

- This option is disabled.
- In cases where either the sending terminal or the target CA WA CA 7 Edition instance are versions before Version 12.0.

If the target CA WA CA 7 Edition instance is using a value for submit class other than the default SUBMIT, that value can optionally be used for the submit check on the sending terminal. The default submit class value is used for either of the following:

- This option is disabled.
- Either the sending terminal or the target CA WA CA 7 Edition instance are versions before Version 12.0.

More information:

SECURITY Statement (see page 26)

U7SVC and External Security

The following information is intended to show the actions that U7SVC performs during execution with relation to security. Two different paths can be taken. The path depends on whether there is an input stream with the U7SVC or if it is only a D= to post a data set.

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 value of SVDSNCHK for the instance not Y, the D= command is passed through to CA WA CA 7 Edition with U7SVC doing no further security checking.

If the value of SVDSNCHK is Y, U7SVC makes a security call. This call determines whether 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 WA CA 7 Edition for processing.

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 whether 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If it is, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value BSUBCHK is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

Data Set Posting

The Batch Card Load Program (SASSBCLP) and U7SVC allow the user to post the creation of a data set to CA WA CA 7 Edition. Because such posting can satisfy requirements or cause job triggering, the need to secure the use of these facilities is critical. Two features of these facilities require a mention in this connection:

- Data set access validation.
- USERID propagation.

The USERID associated with the user of U7SVC or SASSBCLP is extracted to determine the authority of the user to create the data set. Under certain conditions, it is not always 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 initialization file, the USERID associated with the data set creation can be propagated to triggered jobs.

For example, suppose that a user whose USERID is XXX submits a batch job. The batch job uses U7SVC to post the creation of data set A.B to CA WA CA 7 Edition. 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 can be made to verify the authority of the user to create that data set whose creation is for posting to CA WA CA 7 Edition. For more information about the CAIRIM keywords that control this user ID verification, see the chapter "Execution" in the *Systems Programming Guide*.

Sample Definitions

The member AL2ACF2S in the CA WA CA 7 Edition options file (CAL2OPTN) on the installation media contains sample CA ACF2 commands. The sample commands can be used to secure the CA WA CA 7 Edition processing environment under CA ACF2. The definitions are intended as examples only. Review and modify them to meet the security requirements at your location. Once tailored to the specifications of your location, use the definitions as batch input to CA ACF2.

Note: For more information about executing CA ACF2 commands in batch, see the *CA ACF2 Administrator Guide (for z/OS)*.

Chapter 6: Implementing Security with IBM RACF

This chapter describes the steps necessary to implement the CA WA CA 7 Edition External Security Interface with IBM RACF. A working knowledge of the security structure for RACF and its associated command syntax is required. All 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 WA CA 7 Edition security environment. Each site is responsible to determine whether these recommendations meet local auditing and security standards.

This section contains the following topics:

Define Security to RACF

To secure the CA WA CA 7 Edition processing environment under RACF, take the following steps:

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

RACF Requirements

The following topics explain the requirements for security using the RACF product.

System Requirements

Following are the system requirements for RACF:

- RACF must be at a release level supported by IBM.
- To use the CA WA CA 7 Edition External Security Interface with RACF, the CA Standard Security Facility (CAISSF) is required. CAISSF is a subcomponent of the CAIRIM services.

Resource Class Descriptor Table - ICHRRCDE

The Resource Class Descriptor Table is used to identify the general resource classes that RACF is to protect. CA WA CA 7 Edition currently requires two resource classes for implementation of the CA WA CA 7 Edition External Security Interface with RACF. The IBM macro ICHERCDE is used to define the resource classes to the Resource Class Descriptor Table.

Note: For more information about updating the Class Descriptor Table, see the IBM guide *Security Server RACF System Programmer's Guide*.

ICHERCDE Macro

ICHERCDE, the IBM supplied Class Descriptor macro, located in SYS1.MODGEN, defines 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 can reside in any link listed library, however, verify that a copy of this module does not exist in a library anywhere above the new module in the linklist libraries concatenation.

Note: For more information about the ICHRRCDE module and the ICHERCDE macro parameters, see the IBM guide *Security Server RACF System Programmer's Guide*.

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

The resource classes required by CA WA CA 7 Edition are PA@EL and SU@MIT. See 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.

If the CALENDAR option is specified in the EXTERNAL keyword of the SECURITY statement in the initialization file, the resource class of CA@ENDAR is needed. Also, if the RCLASS keyword is used, include the specified value here as a resource class. The CAS9SAFC module probably needs updating to include these classes.

Note: For more information about customizing this module, see the *CA Common Services for z/OS Getting Started*.

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.

Note: For more information about implementing or modifying the RACF Router Table, see the IBM guide *Security Server RACF System Programmer's Guide*.

Add the following entries to the Router Table to implement the CA WA CA 7 Edition External Security Interface with RACF. Add the entries using the IBM ICHRFR01 macro. This module must reside in a linklist library.

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

PA@EL, SU@MIT, and CA@ENDAR

Specifies the label field that 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.

CLASS=CA@ENDAR

Identifies a resource class of CA@ENDAR.

ACTION=RACF

Specifies the action to take for this resource class.

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

Activate the New Resource Classes

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

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

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 authorize access to resources specifically. The Started Procedures Table for RACF allows installations to associate a USERID with a started task that can then be used to specify authorization access.

See the following example:

```
ICHRIN03 CSECT
         TITLE 'ICHRIN03 - STARTED PROCEDURES TABLE'
         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 that 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.

Note: For more information about the Started Procedures Table, see the IBM guide *Security Server RACF System Programmer's Guide*.

Security for the CA WA CA 7 Edition and ICOM started task does not take effect until the USERIDs are defined to RACF.

Define the CA WA CA 7 Edition 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.

This command has the following format:

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

ADDUSER

Identifies the RACF command used to define a new user to RACF.

CA7ONL

Identifies the name chosen in this example for the CA WA CA 7 Edition USERID that is associated with the CA WA CA 7 Edition started task.

NAME('CA 7 ONLINE')

Identifies the RACF parameter used to describe the USERID.

OWNER(CA7GROUP)

Identifies the predefined owning group for the CA WA CA 7 Edition USERID profile. The group or USERID used in the OWNER parameter must already be defined to RACF.

PASSWORD(CA7ONL)

Identifies the password chosen in this example for the CA WA CA 7 Edition USERID. This keyword provides an additional level of security for the CA WA CA 7 Edition started task ID. If no password is specified, the password defaults to the owning group name that is sometimes available to unauthorized personnel.

Define ICOM to RACF

The CA WA CA 7 Edition Independent Communications Manager (ICOM), which manages SMF tracking data for CA WA CA 7 Edition, must also be defined to RACF.

The following example can be used to add the ICOM USERID to RACF:

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

ADDUSER

Identifies the RACF command used to define a new user to RACF.

CA7ICOM

Identifies the name chosen for the ICOM started task.

NAME

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

OWNER

Identifies an existing RACF defined USERID or group that owns this user profile.

PASSWORD

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

CA7ICOM

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

CA 7 and ICOM Data Set Access Requirements

CA WA CA 7 Edition requires access to all of the CA WA CA 7 Edition permanent data sets defined during installation. CA WA CA 7 Edition must also have full access to all JCL libraries that are defined in the initialization file.

Note: For more information about the permanent data sets and the definition of JCL libraries, see the *Systems Programming Guide*.

Define the CA WA CA 7 Edition Application Resource Profile

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

This statement has the following format:

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

RDEFINE

Identifies the command used to define resources to RACF.

APPL

Identifies the resource class name for application resources under RACF.

CA7PROD

Identifies the name chosen in this example for the CA WA CA 7 Edition security application resource name.

DATA

Describes the application resource entry.

OWNER

Identifies an existing RACF defined group that owns the resource.

UACC

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

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

This command has the following format:

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

PERMIT

Identifies the RACF command used to grant access to resources.

CA7PROD

Identifies the name chosen in the previous example for the CA WA CA 7 Edition security application resource name.

CLASS(APPL)

Identifies the resource class for that this command applies. (Application)

ID(*xxxxxxx*)

Identifies the USERID you want to grant access to the CA WA CA 7 Edition security application resource.

Define CA 7 Command and Panel Security to RACF

Security for CA WA CA 7 Edition commands and panels can be protected under RACF by defining each panel and command as a resource. Besides restricting access to top line commands and panels, functions found on each panel can be protected by specifying an access level for each panel. For a list of the CA WA CA 7 Edition commands, panels, and a cross-reference of panel functions with their associated access level requirements, see Security Tables (see page 153).

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

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

RDEFINE

Identifies the RACF command used to define general resources.

PA@EL

Identifies the resource class type for CA WA CA 7 Edition commands and panels. If you have specified a resource type other than PANEL (see the SECURITY statement PCLASS keyword), substitute its value for PA@EL. Also, for resource types other than PANEL, modify the CA Common Services security exit CAS9SAFC.

(L2DB1)

Identifies the resource name for the CA WA CA 7 Edition Job Definition panel.

OWNER(CA7USERS)

Identifies a predefined RACF user or group profile that owns this resource.

UACC(NONE)

Identifies 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.

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

PERMIT

Identifies the RACF command used to grant access to a resource.

L2DB1

Identifies the resource name for the CA WA CA 7 Edition Job Definition panel.

CLASS(PA@EL)

Identifies the resource class type.

ID(*xxxxxxx*)

Identifies the USERID being granted access to the resource.

ACCESS(READ,UPDATE)

Identifies the access level for functions found on the Job Definition panel. The user would have full access to functions that require READ and UPDATE authority.

Define the CA WA CA 7 Edition Agent Job Submission/Command Security

If agent job submissions and command executions are being validated, authorizations are performed to verify that the mainframe user (MFUser) is authorized to submit agent jobs to the specific agent name using the agent user ID. Authorizations are also performed to verify that the signed-on user is authorized to perform any agent command executions. The AGCLASS keyword on the SECURITY statement determines the resource class used for these authorizations.

The following are examples of CA WA CA 7 Edition agent job submission and agent command execution rules:

This example illustrates defining ownership and then giving job submission authority using a specific agent user ID and specific agent name:

```
RDEFINE AGENT (CA71.AGENTUSR) DATA('CA 7 agent job submission') OWNER(CA7USERS)
UACC(NONE)
```

RDEFINE

Identifies the RACF command used to define general resources.

AGENT

Identifies the resource class type for agent job submissions. See the SECURITY statement AGCLASS keyword.

(CA71.AGENTUSR)

Identifies the resource name for CA WA CA 7 Edition agent job submission.

OWNER(CA7USERS)

Identifies a predefined RACF user or group profile that owns this resource.

UACC(NONE)

Identifies the universal access level for this resource. In this case, NONE.

```
PERMIT CA71.AGENTUSR.AGTUSER1.UNIXAGT  CLASS(AGENT) ID(xxxxxxx) ACCESS(READ)
```

PERMIT

Identifies the RACF command used to grant access to a resource.

CA71.AGENTUSR.AGTUSER1.UNIXAGT

Identifies the resource name in the following format:

ca7-instance-id.AGENTUSR.*agent-userid.agent-name*

CLASS(AGENT)

Identifies the resource class type.

ID(*xxxxxxx*)

Identifies the USERID being granted access to the resource.

ACCESS(READ)

Identifies the access level needed for agent job submission.

This example illustrates defining ownership and giving agent command execution authority for a specific agent name:

```
RDEFINE AGENT (CA71.AGENTMSG) DATA('CA 7 agent command execution') OWNER(CA7USERS)
UACC(NONE)
```

RDEFINE

Identifies the RACF command used to define general resources.

AGENT

Identifies the resource class type for agent job submissions. See the SECURITY statement AGCLASS keyword.

(CA71.AGENTMSG)

Identifies the resource name for CA WA CA 7 Edition agent command execution.

OWNER(CA7USERS)

Identifies a predefined RACF user or group profile that owns this resource.

UACC(NONE)

Identifies the universal access level for this resource. In this case, NONE.

```
PERMIT CA71.AGENTMSG.CONTROLSHUTDOWN.UNIXAGT  CLASS(AGENT) ID(xxxxxxx) ACCESS(READ)
```

PERMIT

Identifies the RACF command used to grant access to a resource.

CA71.AGENTMSG.CONTROLSHUTDOWN.UNIXAGT

Identifies the resource name in the following format:

ca7-instance-id.AGENTMSG.*verbsubverb.agent-name*

CLASS(AGENT)

Identifies the resource class type.

ID(*xxxxxxx*)

Identifies the USERID being granted access to the resource.

ACCESS(READ)

Identifies the access level needed for agent command execution.

Secure the /MVS Command

The /MVS command allows a CA WA CA 7 Edition user to issue an MVS console command from a CA WA CA 7 Edition terminal. Although such a facility can prove indispensable in certain situations, the risks associated with an indiscriminate use of the command are obvious. This section discusses security concerns regarding the use of the command.

Note: For more information about the /MVS command, see the *Command Reference Guide*.

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

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

We recommend that you employ CA WA CA 7 Edition command security to restrict /MVS command access to a limited class of privileged users.

Control Job Submission Under RACF

Depending on your security options, CA WA CA 7 Edition can perform submit checking to validate the authorization of a USERID to submit jobs for another USERID. The SECURITY statement describes the options available to perform submit checking.

This example defines a submit (SU@MIT) resource that CA WA CA 7 Edition can validate.

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

RDEFINE

Identifies the RACF command used to define general resources.

SU@MIT

Identifies the resource class type for CA WA CA 7 Edition submission checking. If you have specified a resource type other than SUBMIT (see the SECURITY statement SCLASS keyword), substitute its value for SU@MIT. Also, for resource types other than SUBMIT, modify the CA Common Services security exit CAS9SAFC.

(USERID1)

Identifies the USERID that other user IDs can submit.

DATA

Describes the submission resource class.

OWNER(CA7USERS)

Identifies a predefined RACF user or group profile that owns this resource.

UACC(NONE)

Identifies the universal access level for this resource. In this case, NONE.

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

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

PERMIT

Identifies the RACF command used to grant access to a resource.

USERID1

Identifies the USERID that the ID USERID2 in this example can submit.

CLASS(SU@MIT)

Identifies the resource class type. If you have specified a resource type other than SUBMIT (see the SECURITY statement SCLASS keyword), substitute its value for SU@MIT.

ID(USERID2)

Identifies the USERID that is given submit authority for another ID.

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 WA CA 7 Edition is to submit jobs with USERIDs in the JCL that are different from the CA WA CA 7 Edition USERID, a surrogate designation can be needed. This designation lets CA WA CA 7 Edition submit those jobs with a USERID that is not the same as the one CA WA CA 7 Edition uses.

This method lets CA WA CA 7 Edition submit jobs with various USERIDs, but this method is different from the "Submit Checking" that CA WA CA 7 Edition does. CA WA CA 7 Edition can check for submit authority, but it is done using the SU@MIT class not the SURROGATe class.

Note: For more information about SURROGATe classes, see the IBM guide *Security Server RACF Security Administrator's Guide*.

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

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

PERMIT

Identifies the RACF command used to grant access to a resource.

CLASS(SURROGAT)

Identifies the resource class type.

USERID1.SUBMIT

Identifies the USERID that the ID can submit, CA7ONL in this example.

ID(CA7ONL)

Identifies the USERID that is given submit authority for another ID.

ACCESS(READ)

Identifies the ACCESS level to grant.

Calendar Security

To protect access to the CA WA CA 7 Edition calendars, use the CALENDAR option in the EXTERNAL keyword values on the SECURITY statement in the CA WA CA 7 Edition initialization file.

The Resource Class Descriptor and RACF Router Tables need an entry added for CA@ENDAR before defining any rules for the calendar access. Once these tables are set up to permit CALENDAR security, you can then define the rules.

The following example illustrates the use of the RACF RDEFINE and PERMIT commands first to define the CA WA CA 7 Edition CA@ENDAR as a resource to RACF. Next, the commands *permit* access to specific calendars.

```
RDEFINE CA@ENDAR (calendar-name) DATA('CA 7 Calendar xx') OWNER(CA7USERS)
UACC(NONE)
```

RDEFINE

Identifies the RACF command used to define general resources.

CA@ENDAR

Identifies the resource class type for CA WA CA 7 Edition calendars. If you have specified a resource type other than CALENDAR (see the SECURITY statement CCLASS keyword), substitute its value for CA@ENDAR.

(*calendar-name*)

Identifies the resource name for the calendar that you want to secure. Calendar names have a format of SCAL*yyxx* where *yy* is the year and *xx* are two unique characters identifying the specific calendar.

DATA('CA 7 Calendar *xx*')

Describes the calendar. If the text contains spaces or commas, single quotes are required.

OWNER(CA7USERS)

Identifies a predefined RACF user or group profile that owns this resource.

UACC(NONE)

Identifies the universal access level for this resource. In this case, NONE.

This example grants access to the resource SCAL*yyxx* defined to RACF in the previous example.

```
PERMIT SCALyyxx  CLASS(CA@ENDAR) ID(xxxxxxx) ACCESS(READ)
```

PERMIT

Identifies the RACF command granting access to a resource.

SCAL*yyxx*

Identifies the resource name for the calendar.

CLASS(CA@ENDAR)

Identifies the resource class type. If you have specified a resource type other than CALENDAR (see the SECURITY statement CCLASS keyword), substitute its value for CA@ENDAR.

ID(*xxxxxxx*)

Identifies the USERID granted access to the resource.

ACCESS(READ)

Identifies the access level for the calendar. The only access required is READ, which indicates the user can access the calendar in read or update mode.

More information:

Resource Class Descriptor Table - ICHRRCDE (see page 107)
RACF Router Table (see page 108)

Job Submission

CA WA CA 7 Edition maintains a record of USERIDs that can be associated with a CPU job from queue entry to job submission. This association is not applicable to XPJOB or agent job definitions. XPJOB job and agent job submission security is described in the topic Security Considerations for Cross-Platform Scheduling (see page 18). For agent jobs, also see Define the Agent Submission/Command Security (see page 53).

If requested, a USERID can be inserted into the JCL of a job, before submission, to satisfy batch security requirements on your system. CA WA CA 7 Edition has five potential sources for USERIDs:

Job Owner

Specifies a USERID from the OWNER field for a job on the job definition panel. (SUBUID value of OWNER)

JCL ID

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

Requester

Specifies the USERID of the user that requests a job through the DEMAND, LOAD, or RUN commands. (SUBUID value of REQ)

Queue JCL

Specifies the USERID of a user editing the queued JCL of a job in the CA WA CA 7 Edition request queue. (SUBUID value of QJCL)

CA-7

Specifies the USERID assigned to CA WA CA 7 Edition at startup. If requested, the CA WA CA 7 Edition ID is propagated to submitted jobs. (SUBUID value of CA7)

The specification of the SUBUID keyword on the SECURITY statement in the initialization file determines the priority of USERID sources. The SUBUID keyword specifies a hierarchy of USERIDs for JCL insertion.

At submission time, CA WA CA 7 Edition 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 JCL of a job, and the job is submitted, assuming all other requirements are 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 WA CA 7 Edition checks the status of the SUBNOID flag. The SUBNOID keyword specified on the SECURITY statement in the initialization file sets the SUBNOID flag. If SUBNOID=YES, a job can be submitted without a USERID. If SUBNOID=NO, jobs cannot be submitted without a valid USERID. The jobs 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.

RACF USERID Format

For USERID insertion, CA WA CA 7 Edition 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 a USER= statement follows to supply the USERID. The following example illustrates the JCL statement format used during ID insertion.

```
// USER=userid
```

Note: If a password is required, RACF inserts the USERID password. This function is an automatic function related to Job Submission.

The SUBNOID parameter is used to specify whether a job can 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 the following:

- All candidate USERID sources in the hierarchy were scanned without finding a USERID for the job.
- The SUBNOID parameter requested that jobs are not submitted without a USERID.

Satisfy the R-NOUID Requirement

Two methods can satisfy the R-NOUID requirement.

- If you have specified the QJCL keyword on the SUBUID parameter, you can FETCH/EDIT the queued JCL of the job and immediately do a SAVE/REPLACE. CA WA CA 7 Edition saves the USERID of the user editing the queued JCL. This method 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 add a USERID to the queued JCL of the job manually. CA WA CA 7 Edition recognizes the addition of the USERID to the JCL. The R-NOUID requirement is satisfied.

Note: If the JCL of a job contains a USERID at queue entry time, this USERID overrides all other USERIDs. USERID insertion does not take place.

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

USERID Propagation

Establishing USERIDs to be associated with jobs when submitted by CA WA CA 7 Edition is a critical aspect of security. Defining the USERID hierarchy requires careful planning to verify 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 method 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 that triggers any additional jobs.
- For data set triggers, the USERID associated with a job that CA WA CA 7 Editionsubmitted and created the data set.
- For data set triggers that are initiated through U7SVC and SASSBCLP, the USERID associated with the user posting the creation of the data set.

UID Resources

UID security validation controls access to information about the CA WA CA 7 Edition database. 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 compared to the UID value associated with the job. This comparison provides job-level security for the CA WA CA 7 Edition database. External security does not provide an equivalent JOB level protection. For this reason, it is important that each CA WA CA 7 Edition user is assigned a UID that relates to the user's area of responsibility.

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

The UID value (0-999) can be obtained from the USERID entry in the internal security module, through UID resource validation during logons to CA WA CA 7 Edition or through the /UID top line command issued under a CA WA CA 7 Edition session. If you want to maintain USERIDs in the internal security module, see the internal security chapter for information about 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 that CA WA CA 7 Edition references during UID resource validation. This table contains resource names and associated UID value entries to use during the UID validation process. A sample resource table, SASSRTBL, is provided in both source and load module format and can 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 can change the resource class for calls to external security using the RCLASS= parameter on the SECURITY statement of the initialization file.

You can use the /PROF command to establish and maintain a default UID resource for users logging on to CA WA CA 7 Edition.

Note: For a description of the /PROF command, see the *Command Reference Guide*.

If you use COIDs, the CA WA CA 7 Edition USERID security module described in the USERID macro is required.

CA7RTBL Macro

The CA7RTBL macro is used to generate the UID Resource Table. For the required parameters of this macro, see the following descriptions. Once the new source has been created, see member AL2UM09 in the CA WA CA 7 Edition CAL2OPTN file for applying the USERMOD.

The following is an example of the CA7RTBL macro statement:

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

CA7RTBL

Specifies the UID Resource Table generation macro, which is used to build the UID Resource Table.

RSRC

Defines the resource name to generate in this entry of the table. The resource name can be a one- to eight-character name that meets site-specific external security resource naming conventions.

Resource names must not conflict with existing panel or command names when the default PANEL resource class is used.

Note: For more information, see the description of the RCLASS keyword in the SECURITY statement.

UID

Specifies the value that is associated with the resource name supplied on the RSRC= parameter. The value can be from 0 through 999. 255 is a special UID that grants all access.

Usage Notes

The UID Resource Table name must be a valid PDS member name.

The CA7RTBL macro must be coded starting in column 10.

Duplicate resource names are not permitted, but duplicate UID values are permitted in the table.

The UID Resource Table source must be assembled and link edited into a load library accessible by CA WA CA 7 Edition.

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.

The UID Resource Table name must be identified on the SECURITY statement of the initialization file using the UID= parameter.

The resource names coded in the UID Resource Table must be defined to external security.

/REFRESH Command

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

This command has the following format:

```
REFRESH,MOD=xxxxxxxx
```

MOD=

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

xxxxxxxx

Specifies the member name of the UID Resource Table, and it must reside in a load library accessible to CA WA CA 7 Edition.

UID Resource Table - SASSRTBL Source

The following is a sample 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=CA70255,UID=255                                   00136808
                    .                                                   00137008
         CA7RTBL RSRC=CA70299,UID=299                                   00142808
         CA7RTBL RSRC=CA70300,UID=300                                   00142908
         CA7RTBL RSRC=LAST                                              00143008
         END                                                            00150000
```

CA 7 Logon and UID Resource Validation

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

If the user is defined in the Internal Security module, any UID resource name passed on the Logon panel 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.

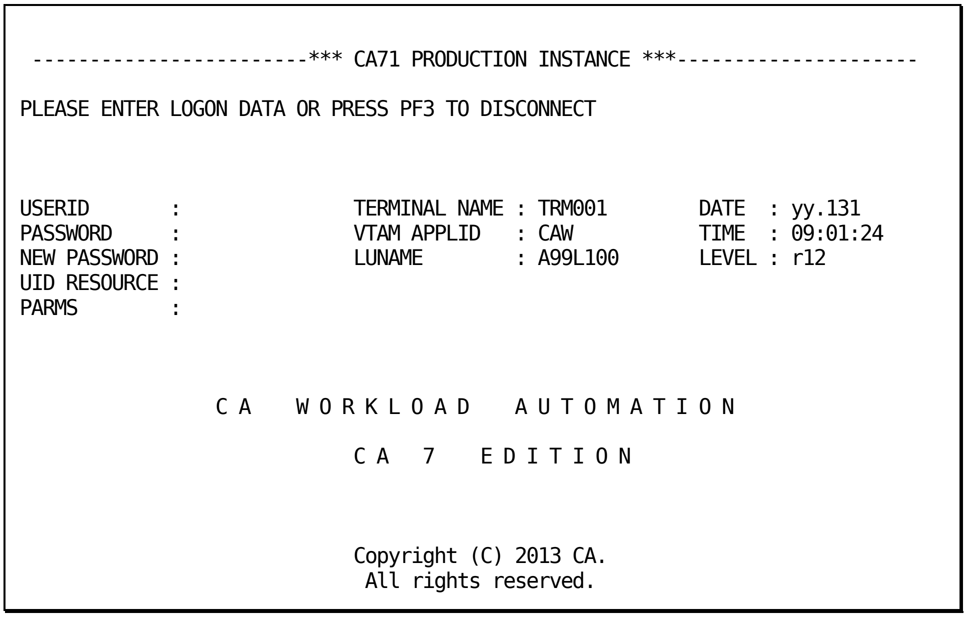
If the resource name is not found in the UID Resource Table, the user is signed on to CA WA CA 7 Edition with a UID value of 0. If a matching resource entry is found, a call is made to external security to validate the authority of the user to access the resource.

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 lets external security control UID assignment to CA WA CA 7 Edition users and eliminates the need to maintain all USERIDs in the CA WA CA 7 Edition Internal Security module.

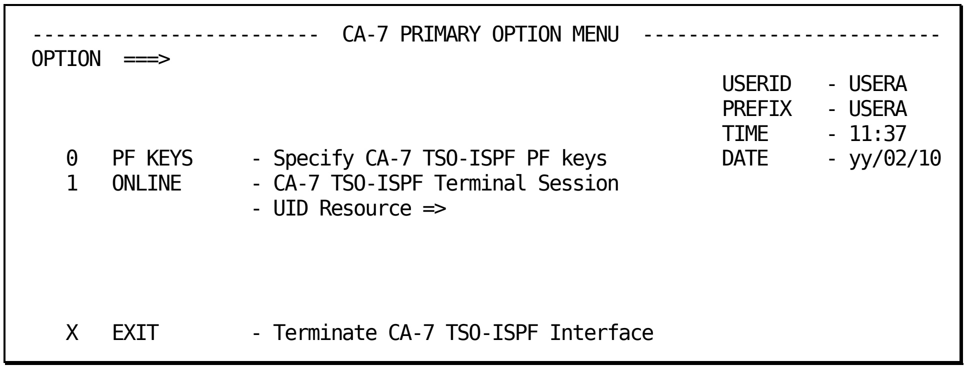
Note: The UID resource validation process is the same under the CA WA CA 7 Edition ISPF Interface; however, no password is required.

The following is a sample CA WA CA 7 Edition Logon panel:



```
 ------------------------*** CA71 PRODUCTION INSTANCE ***---------------------

PLEASE ENTER LOGON DATA OR PRESS PF3 TO DISCONNECT



USERID       :              TERMINAL NAME : TRM001       DATE  : yy.131
PASSWORD     :              VTAM APPLID   : CAW          TIME  : 09:01:24
NEW PASSWORD :              LUNAME        : A99L100      LEVEL : r12
UID RESOURCE :
PARMS        :



                 C A   W O R K L O A D   A U T O M A T I O N

                             C A   7   E D I T I O N



                             Copyright (C) 2013 CA.
                              All rights reserved.
```

The following is a sample CA WA CA 7 Edition ISPF Interface Primary Option Menu panel:



```
 -------------------------  CA-7 PRIMARY OPTION MENU  -------------------------
 OPTION  ===>
                                                           USERID   - USERA
                                                           PREFIX   - USERA
                                                           TIME     - 11:37
    0   PF KEYS     - Specify CA-7 TSO-ISPF PF keys        DATE     - yy/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 WA CA 7 Edition TSO-ISPF.

/UID Command

The /UID command can change the current UID processing value of a user through UID resource validation. The /UID command requires the following:

- The UID Resource Table option is implemented.
- The resources and the appropriate security authorization are defined to external security.

This command has the following format:

```
/UID,{R=resname|LIST}
```

R=*resname*

Identifies a resource name that exists in the UID Resource Table. The authorization of the user to access the resource is validated through external security. If authorized, the current UID value of the user is updated. The updated value reflects 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.

Program Protection

The following topics discuss program protection for CA WA CA 7 Edition and batch users.

CA 7 Requirements

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

Batch Users

Batch USERIDs that are associated with jobs submitted through CA WA CA 7 Edition require access to the program SASSJJCL. SASSJJCL is the LOAD program that identifies resources used by a job to CA WA CA 7 Edition.

To prevent the unauthorized use of CA WA CA 7 Edition, we recommend that you secure access to the following programs:

- SASSBCLP - Batch Card Load Program
- SASSBSTR - Batch Terminal Interface
- SASSTRLR - Trailer Step Facility
- U7SVC - CA WA CA 7 Edition SVC Facility
- CAL2X2W0 - CA WA CA 7 Edition CAICCI Interface
- CAL2X2T0 – CA WA CA 7 Edition TCP/IP Interface

Group Profiles

RACF lets you define Group Profiles that are used to organize security access at a group level rather than by individual users. Although support group identification at logon is not supported, you can structure your resource authorizations by group and then "connect" users to the specific Group Profiles. If you use multiple "connect groups," activate the List-of-Groups authorization checking feature in RACF.

Note: For more information about Group level access, see the *RACF Security Administrators Guide*.

External Communicators with IBM-RACF

The external communicators (SASSBSTR, SASSTRLR, U7SVC, CAL2X2W0, CAL2X2T0, and SASSBCLP) provide a means for users outside the CA WA CA 7 Edition address space to communicate with CA WA CA 7 Edition. Because the use of these programs sometimes permits access to production jobs, we recommend that you give careful consideration 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 can be implemented to prevent unauthorized use of these facilities. This section describes those controls.

Users of the CA WA CA 7 Edition CAICCI Interface require access to the program CAL2X2W0. This requirement is true regardless of whether this interface is invoked in batch (program CAL2X2WB), REXX (program CAL2X2WR), or program-to-program (CAL2X2WP).

Users of the CA WA CA 7 Edition TCP/IP interface require access to the program CAL2X2T0. This requirement is true regardless of whether this interface is invoked in batch (program CAL2X2TB), REXX (program CAL2X2TR), or program-to-program (CAL2X2TP).

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

- Terminal communication (Batch Terminal Interface, Trailer, CA WA CA 7 Edition CAICCI Interface, CA WA CA 7 Edition TCP/IP Interface, and U7SVC)
- Data set posting (U7SVC and SASSBCLP)

Terminal Communication

Each of the following let the user send terminal commands to CA WA CA 7 Edition:

The Batch Terminal Interface (SASSBSTR), the Trailer facility (SASSTRLR), the CA WA CA 7 Edition CAICCI Interface (CAL2X2W0), CA WA CA 7 Edition TCP/IP Interface (CAL2X2T0), and U7SVC

Although no online terminal is used with this mode of communication, input from these programs is treated as terminal input by CA WA CA 7 Edition. Command security in these environments is handled as it is for all CA WA CA 7 Edition terminals. IBM-RACF controls access to CA WA CA 7 Edition commands if EXTERNAL=COMMAND is specified on the SECURITY statement in the initialization file. IBM-RACF determines the access of a user to CA WA CA 7 Edition terminal commands based on the USERID supplied on the /LOGON command. Thus, when using an External Communicator, a /LOGON command must precede any command input.

IBM-RACF typically requires a password at logon. But including passwords in command input for the External Communicators would obviously represent a serious security exposure. Several checks can 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, a /LOGON statement is built using the USERID associated with the current user. Under certain conditions, it is not always 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 can 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 can validate the READ access of the user to an entity whose name is the USERID found on the /LOGON statement. The resource class is SU@MIT. Create security definitions for this resource 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 WA CA 7 Edition allows the user to LOGON without a password.

The USERID of the current user is determined using CAS9 CAISSF services.

Note: For more information about CAISSF, see the CA Common Services documentation.

The value of BSUBCHK that CAIRIM sets controls submit checking for External Communicators.

Note: For more information, see the chapter "Execution" in the *Systems Programming Guide*.

SASSTRLR and External Security

The following information shows the actions that SASSTRLR performs 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 whether 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If so, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value of BSUBCHK for the instance is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y, and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

SASSBSTR, CAL2X2W0, CAL2X2T0, and External Security

The following information shows the actions that SASSBSTR, CAL2X2W0, and CAL2X2T0 perform during execution with relation to security.

A security EXTRACT is done to determine the USERID of the submitted job or user environment. 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 whether a logon statement was supplied. If no logon statement was supplied, 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If so, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value of BSUBCHK for the instance is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y, and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

For CAL2X2W0 (CAICCI Terminal) and CAL2X2T0 (TCP/IP Terminal) executions, the BSUBCHK setting for the target instance of CA WA CA 7 Edition can optionally be used.

If used, the setting overrides any setting on the submitting terminal. The submitting terminal BSUBCHK setting is used for either of the following:

- This option is disabled.
- In cases where either the sending terminal or the target CA WA CA 7 Edition instance are versions before Version 12.0.

If the target CA WA CA 7 Edition instance is using a value for submit class other than the default SUBMIT, that value can optionally be used for the submit check on the sending terminal. The default submit class value is used for either of the following:

- This option is disabled.
- Either the sending terminal or the target CA WA CA 7 Edition instance are versions before Version 12.0.

More information:

SECURITY Statement (see page 26)

U7SVC and External Security

The following information is intended to show the actions that U7SVC performs during execution with relation to security. Two different paths can be taken. The path depends on whether there is an input stream with the U7SVC or if it is only a D= to post a data set.

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 value of SVDSNCHK for the instance not Y, the D= command is passed through to CA WA CA 7 Edition with U7SVC doing no further security checking.

If the value of SVDSNCHK is Y, U7SVC makes a security call. This call determines whether 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 WA CA 7 Edition for processing.

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 whether 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 WA CA 7 Edition 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 WA CA 7 Edition 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 whether the value of BSUBCHK for the instance is Y. If it is, a submit check is performed. The check is done to see whether 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 WA CA 7 Edition indicating that no password is needed with this particular logon attempt.

If the value BSUBCHK is not Y, no submit checks are done. The logon statement is passed as it is coded with no special indication to CA WA CA 7 Edition. If CA WA CA 7 Edition 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, the external security package validates it. If no password was coded, the logon fails due to a missing password.

In general, a password is needed only two times:

- The user codes the user exit SASSXXLX to require a password.
- The value of BSUBCHK is not Y and the OPID is different from the EXTRACTed ID.

Note: Values of SVDSNCHK and BSUBCHK are set using CAIRIM. This method is discussed in the chapter "Execution" in the *Systems Programming Guide*. There you can also find information about CAL2ENVR, a utility that reports current options used by each instance of CA WA CA 7 Edition supported on the LPAR. CAL2ENVR can be used to determine the current settings of keywords such as SVDSNCHK and BSUBCHK.

Data Set Posting

The Batch Card Load Program (SASSBCLP) and U7SVC allow the user to post the creation of a data set to CA WA CA 7 Edition. Because such posting can satisfy requirements or cause job triggering, the need to secure the use of these facilities is critical. Two features of these facilities require a mention in this connection:

- Data set access validation.
- USERID propagation.

The USERID associated with the user of U7SVC or SASSBCLP is extracted to determine the authority of the user to create the data set. Under certain conditions, it is not always 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 initialization file, the USERID associated with the data set creation can be propagated to triggered jobs.

For example, suppose that a user whose USERID is XXX submits a batch job. The batch job uses U7SVC to post the creation of data set A.B to CA WA CA 7 Edition. 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 can be made to verify the authority of the user to create that data set whose creation is for posting to CA WA CA 7 Edition. For more information about the CAIRIM keywords that control this user ID verification, see the chapter "Execution" in the *Systems Programming Guide*.

Sample Definitions

The member AL2RACFS in the CA WA CA 7 Edition options file (CAL2OPTN) on the installation media contains sample RACF commands. The sample command can be used to secure the CA WA CA 7 Edition processing environment under RACF. The definitions are intended as examples only. Review and modify them to meet the security requirements at your location. Once tailored to the specifications of your location, use the definitions as batch input to RACF.

Note: For more information about executing RACF commands in batch, see the *Security Server RACF Command Language Reference*.

Chapter 7: Internal Security

This section contains the following topics:

Overview

The user can define the security structure and level of authority for each individual allowed access to CA WA CA 7 Edition. Make a careful evaluation of security requirements before implementing CA WA CA 7 Edition.

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

- Terminal/Operator
- Operator/CA WA CA 7 Edition Application
- CA WA CA 7 Edition 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 definition means each level of security defined by the SECURITY macro requires that preceding levels are also defined. For example, to have authorization to perform a specific command within an application (Application/Command security), an operator requires 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 WA CA 7 Edition.

Establishing the user-defined security structure is accomplished through the CA WA CA 7 Edition initialization process. The CA WA CA 7 Edition 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 WA CA 7 Edition, which terminal each operator can use, and other authorization qualifiers necessary to limit the interface with CA WA CA 7 Edition. The definition of control provided can be modified whenever necessary because of changes in personnel, addition/deletion of terminals, and so forth. Because the initialization file pointing to this matrix is reloaded each time CA WA CA 7 Edition is initialized, the most current definition of the data center's security structure is always in effect.

Security Use Considerations

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

- The Master Terminal Operator (MTO) has the most responsibility for the activity of CA WA CA 7 Edition. MTO functions can include the following:

  - Monitoring production activity
  - Providing status reports
  - Responding to CA WA CA 7 Edition's requests for JCL overrides or manual verifications
  - Using various application functions to help ensure database integrity.

  To permit the MTO to perform these various functions, the SECURITY macro statement of the MTO would allow access to all functions within all CA WA CA 7 Edition 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 can have different levels of authorization depending on which terminal is being used and which functions are being performed.

- Database Maintenance can be established as a separate activity that one person or a designated group of personnel can control and perform. 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 can require access to CA WA CA 7 Edition in inquire-only mode. For example, RJE users could be authorized to interrogate the queues to determine job status. The users can 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 WA CA 7 Edition for information reflecting current production status, projected schedules, history, and so forth.

Regardless of the level of responsibility as defined to CA WA CA 7 Edition, all authorized operators (MTO through end user) must use the same procedure to establish communication with CA WA CA 7 Edition. This procedure is the logon and is described in the *Command Reference Guide*.

Define Security Levels

Once the requirements are defined, review and plan the following two general areas of activity:

- SECURITY macros generate the load module referenced in the initialization file. The default Security module distributed with CA WA CA 7 Edition does not restrict security.
- If you identified the need for joint ownership of classified jobs or external data sets, establish UIDs with an optional USERID Security module.

The following topics provide a description of each of the five levels of security.

Terminal/Operator Security

To enforce the security structure defined, CA WA CA 7 Edition requires identification of all personnel (terminal operators). A unique operator ID identifies each terminal operator, up to eight characters, which is specified in the SECURITY macro. The macro must also specify the terminal IDs of each terminal on which this operator is allowed to log on (/LOGON).

Operator/Application Security

An operator can be restricted to the use of only those applications that fall within the specific area of responsibility of the operator. The SECURITY macro defines access to applications for each operator. Application security levels are shown in the table in Application/Command Security.

Application/Command Security

Within each CA WA CA 7 Edition application is at least one command that an operator can perform. Each command is assigned a required authorization value (level) from 0 to 15 in the SASSTRAN module. You can modify this program to adjust these values. To disable any command, assign it a level greater than 15 in SASSTRAN. Commands assigned lower numbers are less restrictive than commands assigned higher numbers. For example, inquiry commands can have level 4, while utilities can 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.

Application ID	Function Authority Level	Function Name
AR0 (Automated Recovery Facility)	00	AR.3
MLR (Management Level Reporting)	00	GRAPHD, GRAPHJ, GRAPHN, GRAPHS
PS0 (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	PRRNDEL, PRRNJCL, RESANL, RQVER
SCM (System Commands)	00	/AFM, /CLOSE, /CONT, /COPY, /DISPLAY, /ECHO, /FETCH, /NXTMSG, /PAGE, /PA*nn*, /PF*nn*, /PROF
	01	/OPERID, /UID
	05	/AUTO, /EMAIL, /LOG, /MSG, /SDESK
	10	/BRO, /DRCLASS, /PURGPG, /WTO
	12	/DRMODE, /EADMIN, /START, /STOP, /SWAP
	13	/RESET
	14	/ASSIGN, /CHANGE, /CLOSE(T=), /DMP1, /DUMP, /GVAR, /LOGOFF(T=), /MVS, /OPEN, /RELINK, /REFRESH, /SHUTDOWN, /WLB, /XCF, /XTASK
	15	/AGENT, /COID, /DELAGNT, /IAS, /JCL, /OPERIDS, /PROFS, /STATEMGR

Application ID	Function Authority Level	Function Name
SDM (Database Maintenance)	00	AGJOB, CALMOD, DBM, DSN, PROSE, PROSE(DD), PROSE(DSN), PROSE(JOB), PROSE(NETWORK), PROSE(JOB), PROSE(USER), JCL, JOB, JOBCONN, JOBCONN(DSN), JOBCONN(JDEP), JOBCONN(JOBL), JOBCONN(NWK), JOBCONN(NWKL), JOBCONN(USR), JOBCONN(USRL), NETWORK, QJCL, SCHD, SCHD(DTRG), SCHD(DTRGL), SCHD(INWK), SCHD(JOB), SCHD(JTRG), SCHD(JTRGL), SCHD(NTRG), SCHD(NTRGL), SCHD(ONWK), SCHDMOD, XPJOB (See SASSDSCR source module for function security levels)
	10	CONVERT, RESTORE
	15	AGPSWD, XNODE, XPSWD
SFC (Forecast)	04	FALL, FJOB, FPOST, FPRE, FQALL, FQJOB, FQPOST, FQPRE, FQRES, FQSTN, FQTAPE, FRES, FRJOB, FRQJOB, FSTN, FSTRUC, FTAPE, FWLP
SJR (Job Restart)	10	LIST, RESTART
SLI (Inquiry and Report)	00	HELP
	01	FLOWL, LACT(R), LAGENT, LARF, LARFQ, LCTLG, LDSN, LDTM, LJES, LJOB(R), LLOCK, LNODE, LNTWK, LOC, LPOST, LPRE, LPRRN, LPROS, LQ(P/R), LRDY(P/R), LREQ(P/R), LRES, LRLOG, LRMD, LSCHD, LSYS, LVAR, LWLB, LXCF
	04	LGVAR, 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, FLOWD, HOLD, JCLOVRD, JOBSTART, LOAD(H), NOPRMP, NXTCYC, POST, PRMP, PRSCF, PRSQA, PRSQD, RELEASE, REPLY, REQUEUE, RESCHNG, RUN(H), RUNNW, RUSH, SUBMIT, SUBSCH, SUBTM, VERIFY
	15	MOVE, SSCAN, START, STOP, X
SQM (Queue Maintenance)	00	XPOST, XPRE, XQ, XQJ, XQM, XQN, XRQ, XRST, XSPOST, XSPRE, XUPD, XWLB, (See SASSDSCR source module for function security levels)
	15	AGFILE

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

Command/Function Panel 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 panel, function and terminal.

The security table is defined in SASSDSCR. The source for SASSDSCR is distributed in library CAL2SRC. It is composed of SEC macros, which are described here:

```
SEC SCR=nnn,PAN=nnnnnnnn
   [,ADD={0|nn}]
   [,DEL={0|nn}]
   [,READ={0|nn}]
   [,SUBM={0|nn}]
   [,UPD={0|nn}]
   [,TERMLVL=nn]
   [,TERM=(termn,...)]
```

SCR=*nnn*

Identifies the panel. Required. Do not change this value.

PAN=*nnnnnnnn*

Identifies the panel ID. Required. Do not change this value.

ADD=*nn*

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

DEL=*nn*

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

READ=*nn*

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

SUBM=*nn*

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

UPD=*nn*

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

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

TERMLVL=*nn*

(Optional) Identifies the authorization level (from 0 to 15) at which the terminal list is examined for access qualification. This setting lets you restrict certain panel functions like UPD or DELETE to specific terminals but permit functions like LIST without terminal restrictions.

TERM=(*term1*,...,*termn*)

(Optional) Identifies one or more terminals to validate when examining function access. The maximum number of characters that can be within the parentheses is 255. If this limit is encountered, specify as many terminal names as can 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, the user is allowed to execute all functions.
- In all other cases, the authorization level as defined in the Security module is compared against the following:
  - The security level required for the appropriate panel.
  - The function (READ, ADD, and so forth).

  If the authorization level is less, the command is rejected.
- The TERM list is now examined for terminal restrictions. If no restrictions are found, the command is accepted. If the security level required is less than the TERMLVL value, the command is accepted. Otherwise, the TERM list is examined. If the terminal of the user is not found in this list, the command is rejected.

If EXTERNAL=COMMAND was specified on the SECURITY statement in the CA WA CA 7 Edition initialization file, do not modify SASSDSCR. If CA WA CA 7 Edition native security controls command access, you can modify SASSDSCR to reflect the needs of your installation.

For a sample SMP/E USERMOD to modify the SASSDSCR security table, see member AL2UM04 in the CA WA CA 7 Edition Options library CAL2OPTN.

UID/External Data Set Security

The UID/External Data Set security option helps ensure that only authorized personnel have access to specific classified jobs and identified data sets external to CA WA CA 7 Edition. 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 WA CA 7 Edition. 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 SECURITY macro definition of the operator must specify one of the following:

- A UID matching the UID of the job.
- The data set defined in the USERID Security module.

Because 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.

SECURITY Macro

The SECURITY macro defines the access levels for all personnel who are designated as CA WA CA 7 Edition operators as follows:

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

A Security module should be assembled and link edited for the specific environment of the user. The module must not be linked as reentrant, reusable. CA WA CA 7 Edition 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 WA CA 7 Edition, this security is in effect. A sample Security module resides on the CA WA CA 7 Edition Source library with the name SASSSECI.

This macro has the following format:

```
name SECURITY APLID=(xxx0,nn),OPID=xxxxxxxx,TRM=xxxxxxx
   [,USRID={0|nnn}]
   [,LAST=YES]
```

name

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

APLID

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

***xxx*0**

Identifies the CA WA CA 7 Edition application driver (must end in 0). For the first three characters of each application ID, see the table in the Application/Command Security topic.

nn

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

A sublist in the following format can specify multiple applications:

((*xxx*0,*nn*),...,(*xxx*0,*nn*))

OPID

Specifies an identification code that an operator must use when logging on to a CA WA CA 7 Edition terminal. The value must be alphanumeric, up to eight characters. OPID is required and has no default. More than one OPID can be specified in the following format:

OPID=(*xxxxxxxx,...,xxxxxxxx*)

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

TRM

Specifies the names, in up to seven characters, of the CA WA CA 7 Edition terminals that the designated operators 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 has no default. More than one terminal name can be specified using a sublist notation of TRM=(*xxxxxxx,...,xxxxxxx*). A value of **ALL** propagates the specified operator definitions to all terminals defined in the initialization file. At least one TRM=**ALL** must be specified to use the Virtual Terminal feature.

USRID

(Optional) Specifies a user (ownership) identification that controls the ability of the operator to access information in the database. Value must be a number from 0 through 999. The default is 0. 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. We recommend that you place a PRINT NOGEN statement before the first SECURITY macro to suppress the macro expansion printout, which can be lengthy.

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 can perform the following:

- 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 panel).
- Enter database maintenance commands (SDM0) requiring an authority level of 3 or less.
- Enter queue maintenance command (SPO0) requiring an authority level of 9 or less.

Terminal TERM3 can be logged on with OPIDs of OP1, OP2, or OP3. The operators can perform the following:

- 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.

USERID Macro

The USERID macro defines the correspondence, if any, between various UIDs for access to classified jobs, and can also identify data sets external to CA WA CA 7 Edition 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 WA CA 7 Edition 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 WA CA 7 Edition UID feature.

To display information about the user IDs and their correspondence IDs (COIDs), the /COID command may be used.

Note: For more information about this command, see the *Command Reference Guide*.

Two formats are associated with the USERID macro. One is for ID correspondence, and the other is for data set protection. All USERID macros using UID keywords *must* precede the first DSN entry or assembly errors will result.

For ID correspondence, the macro has the following format:

```
name USERID {UID|U}=(nnn)
                    (nnn,...,nnn)
                    (nnn-nnn)
                    (nnn-nnn,,,nnn)
   [,{COIDS|C}=(nnn)         ]
   [          (nnn,...,nnn)  ]
   [          (nnn-nnn)      ]
   [          (nnn-nnn,,,nnn)]
   [,LAST=YES]
```

For external data set protection, the macro has the following format:

```
name USERID {DSNAME|DSN|D}=(xxxx,....)
   [,{READ|R}=(ALL)          ]
   [          (nnn)          ]
   [          (nnn,...,nnn)  ]
   [          (nnn-nnn)      ]
   [          (nnn-nnn,,,nnn)]
   [          (WRITE)        ]
   [,{WRITE|W}=(ALL)         ]
   [          (nnn)          ]
   [          (nnn,...,nnn)  ]
   [          (nnn-nnn)      ]
   [          (nnn-nnn,,,nnn)]
   [          (READ)         ]
   [,MBROPT={R|W|RW|WR}]
   [,LAST=YES]
```

name

(Optional) Specified only on the first USERID statement in the module to generate the CSECT name for the module. The default is UIDTABLE.

Start the macro name USERID 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 can be used interchangeably. The value can be a single ID, a range of IDs or a sublist combining the previous two values. IDs can be any value from 0 through 999 in ascending order. 255 is a special value reserved for all access and cannot be specified. It is automatically generated. Specifying a value of 255, either by itself or within a range, generates an error at assembly time.

C|COIDS

Specifies UIDs that the UID inherits access. (This access is for job access only, not external data set access, unless MBROPT is used.) Can 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 protected with UID security. DSNAME, DSN and D can be used interchangeably. The value is a data set name or names up to 44 characters each, which can be enclosed in quotes. Indicate a generic name by ending it with an asterisk (*), like SYS*. This keyword cannot be specified on a statement containing UID. If DSNAME is used with no other keywords, 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 allow read access to data sets in the DSN list in ascending order. READ and R can be used interchangeably. This keyword has the same format as UID. Can 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 cannot be specified on a statement containing the UID keyword. If MBROPT is used, READ is ignored.

W|WRITE

Specifies a UID or list or ranges in ascending order to allow write access to data sets in the DSN list. WRITE and W can be used interchangeably. This keyword has the same format as UID. Can 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 cannot be specified on a statement containing the UID keyword. If MBROPT is used, 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 WA CA 7 Edition database. Cannot be used on statements containing the UID keyword. Values can 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, 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. Code this keyword only on the last macro statement of the module. The value must be YES.

An assembler END statement must appear after the last USERID macro. We recommend that you place a PRINT NOGEN statement before the first USERID macro to suppress the macro expansion.

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
```

Appendix A: Security Tables

CA WA CA 7 Edition includes several security tables. They are described in the following topics.

This section contains the following topics:

Panel-ID Table

The following table lists the panel-IDs. Each panel-ID has a corresponding resource name to use when defining the resource rules for securing CA WA CA 7 Edition. An asterisk identifies any panel-ID that is new for this release.

Note: Use the resource name listed in the table for each panel when defining the resource rules under external security. The L2 in the resource name is the CA WA CA 7 Edition product code and is required.

Panel-ID	Resource Name	Description	New in Version 12.0.00
APA	L2AP	Automated Performance Analysis Menu (APA)	
AP.1	L2AP1	Automated Performance Analysis Prompt	
AP.2	L2AP2	Automated Performance Analysis Prompt	
AP.3	L2AP3	Automated Performance Analysis Prompt	
AP.4	L2AP4	Automated Performance Analysis Prompt	
AP.5	L2AP5	Automated Performance Analysis Prompt	
AR	L2AR	ARF	
AR.3	L2AR3	ARF Condition Definition Maintenance	
DB	L2DB	Database Maintenance Menu (DBM)	
DB.A	L2DB1	DBM - Cross Platform Jobs Menu	
DB.A.A	L2DB1	DBM - XPJOB	

Panel-ID	Resource Name	Description	New in Version 12.0.00
DB.A.B	L2DB1	DBM - UNIX	
DB.A.C	L2DB1	DBM - Windows	
DB.A.D	L2DB1	DBM - File Trigger (Watch)	
DB.A.E	L2DB1	DBM - FTP Jobs Menu	
DB.A.E.A	L2DB1	DBM - FTP	
DB.A.E.B	L2DB1	DBM - Secure Copy (SCP)	
DB.A.E.C	L2DB1	DBM - Secure File Transfer (SFTP)	
DB.A.F	L2DB1	DBM - SAP Jobs Menu	
DB.A.F.A	L2DB1	DBM - Batch Input Session (BDC)	
DB.A.F.B	L2DB1	DBM - Business Warehouse InfoPackage (BWIP)	
DB.A.F.C	L2DB1	DBM - Business Warehouse Process Chain (BWPC)	
DB.A.F.D	L2DB1	DBM - SAP Job	
DB.A.F.E	L2DB1	DBM - SAP Archive Job	
DB.A.F.F	L2DB1	DBM - SAP Event Monitor	
DB.A.F.G	L2DB1	DBM - SAP Process Monitor	
DB.A.G	L2DB1	DBM - PeopleSoft	
DB.A.H	L2DB1	DBM - ORACLE Jobs Menu	
DB.A.H.A	L2DB1	DBM - Oracle Request	
DB.A.H.B	L2DB1	DBM - Oracle Copy	
DB.A.I	L2DB1	DBM - Object Monitor Menu	
DB.A.I.A	L2DB1	DBM - CPU Monitor	
DB.A.I.B	L2DB1	DBM - Disk Monitor	
DB.A.I.C	L2DB1	DBM - IP Monitor	
DB.A.I.D	L2DB1	DBM - Process Monitor	
DB.A.I.E	L2DB1	DBM - Text File Monitor	
DB.A.I.F	L2DB1	DBM - Event Log Monitor	
DB.A.I.G	L2DB1	DBM - Service Monitor	
DB.A.J	L2DB1	DBM - Data Base Jobs Menu	
DB.A.J.A	L2DB1	DBM - SQL	

Panel-ID	Resource Name	Description	New in Version 12.0.00
DB.A.J.B	L2DB1	DBM - Stored Procedure	
DB.A.J.C	L2DB1	DBM - Monitor	
DB.A.J.D	L2DB1	DBM - Trigger	
DB.A.K	L2DB1	DBM - OS400	
DB.A.L	L2DB1	DBM - Java Jobs Menu	
DB.A.L.A	L2DB1	DBM - J2EE JMS Publish	
DB.A.L.B	L2DB1	DBM - J2EE JMS Subscribe	
DB.A.L.C	L2DB1	DBM - J2EE Entity Bean	
DB.A.L.D	L2DB1	DBM - J2EE HTTP/Servlet	
DB.A.L.E	L2DB1	DBM - J2EE POJO	
DB.A.L.F	L2DB1	DBM - J2EE RMI	
DB.A.L.G	L2DB1	DBM - J2EE Session Bean	
DB.A.L.H	L2DB1	DBM - JMX-Mbean Attribute Get	
DB.A.L.I	L2DB1	DBM - JMX-Mbean Attribute Set	
DB.A.L.J	L2DB1	DBM - JMX-Mbean Operation	
DB.A.L.K	L2DB1	DBM - JMX-Mbean Subscribe	
DB.A.L.L	L2DB1	DBM - JMX-Mbean Create Instance	
DB.A.L.M	L2DB1	DBM - JMX-Mbean Remove Instance	
DB.A.M	L2DB1	DBM - SNMP Jobs Menu	
DB.A.M.A	L2DB1	DBM - Get Attribute	
DB.A.M.B	L2DB1	DBM - Set Attribute	
DB.A.M.C	L2DB1	DBM - Subscribe	
DB.A.M.D	L2DB1	DBM - Trapsend	
DB.A.N	L2DB1	DBM - Web Services	
DB.A.O	L2DB1	DBM - Wake-on-LAN	
DB.A.P	L2DB1	DBM - Remote Execution	*
DB.A.Q	L2DB1	DBM - HP Integrity NonStop	
DB.1	L2DB1	DBM - CPU Job Definition	
DB.10	L2DB1	DBM - XPJOB Definition	

Panel-ID	Resource Name	Description	New in Version 12.0.00
DB.11	L2DB1	DBM - Agent 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.2.4L	L2DB24	DBM – Job Triggering (Long Job Name)	*
DB.2.5	L2DB25	DBM - Input Network Triggering	
DB.2.5L	L2DB25	DBM – Input Network Triggering (Long Job Name)	*
DB.2.6	L2DB26	DBM - Data Set Triggering	
DB.2.6L	L2DB26	DBM – Data Set Triggering (Long Job Name)	*
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	
DB.3.1	L2DB31	DBM - Data Set Predecessors	
DB.3.2	L2DB32	DBM - CPU Job Predecessors	
DB.3.2L	L2DB32	DBM – CPU Job Predecessors (Long Job Name)	*
DB.3.4	L2DB34	DBM - Input/Output Network Tasks	
DB.3.4L	L2DB34	DBM – Input/Output Network Tasks (Long Job Name)	*
DB.3.6	L2DB36	DBM - User Memo-Form Predecessors	
DB.3.6L	L2DB36	DBM – User Memo-Form Predecessors (Long Job Name)	*
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	

Panel-ID	Resource Name	Description	New in Version 12.0.00
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 - Job 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	
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	

Panel-ID	Resource Name	Description	New in Version 12.0.00
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.6-XF	L2QM6F	QM - Input Networks - Free	
QM.6-XH	L2QM6H	QM - Input Networks - Hold	
QM.6-XI	L2QM6I	QM - Input Networks - Login	
QM.6-XO	L2QM6O	QM - Input Networks - Logout	
QM.6-XP	L2QM6P	QM - Input Networks - Respond to Prompting	
QM.6-XR	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)	
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	

Panel-ID	Resource Name	Description	New in Version 12.0.00
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	Utilities	
UT.1	L2UT1	Utilities - Allocate Data Set	
UT.1C	L2UT1C	Utilities - Allocate/Catalog Data Set	
UT.10	L2UT10	Utilities - Find Data Set on DASD	
UT.11	L2UT11	Utilities - Allocate Volume	
UT.12	L2UT12	Utilities - Deallocate Volume	
UT.13	L2UT13	Utilities - Display Format 1 DSCB	
UT.14	L2UT14	Utilities - Display Directory Information	
UT.15	L2UT15	Utilities - Display Data Set Attributes Map	
UT.16	L2UT16	Utilities - Display Available DASD Space	
UT.17	L2UT17	Utilities - Display Physical Data Records	
UT.18	L2UT18	Utilities - Display Catalog Block	
UT.19	L2UT19	Utilities - Display Catalog Entries	
UT.2	L2UT2	Utilities - Catalog Data Set	
UT.3	L2UT3	Utilities - Rename Data Set	

Panel-ID	Resource Name	Description	New in Version 12.0.00
UT.4	L2UT4	Utilities - Scratch Data Set	
UT.5	L2UT5	Utilities - Uncatalog Data Set	
UT.6	L2UT6	Utilities - Build GDG Index	
UT.7	L2UT7	Utilities - Delete Index	
UT.8	L2UT8	Utilities - Connect a Catalog	
UT.9	L2UT9	Utilities - Disconnect a Catalog	
WB.X	L2WBX	Workload Balancing Maintenance	
XN.1	L2DBXNOD	DBM - XPJOB Node Definition	
XN.2	L2DBXPSW	DBM - XPJOB User-ID/Password Definition	
XN.3	L2DBAPSW	DBM - Agent User-ID/Password Definition	

Command Table

The following table lists the commands and the corresponding resource name to use when defining the resource rules to external security. The service level is READ. The L2 prefix on the resource names is the CA WA CA 7 Edition 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.

Note: For more information about the following commands, see the *Command Reference Guide*:

Command	Resource Name	New in Version 12.0.00
/AFM	L2SCAFM	
/AGENT	L2SCAGNT	
/ASSIGN	L2SCASSI	
/AUTO	L2SCAUTO	
/BRO	L2SCBRO	
/CHANGE	L2SCCHAN	
/CHGOPT	L2SCCOPT	*
/CLOSE	N/A	

Command	Resource Name	New in Version 12.0.00
/CLOSE(T)	L2SCCLOS	
/COID	L2SCCOID	
/CONT	N/A	
/COPY	N/A	
/DELAGNT	L2SCDELA	
/DISPLAY	N/A	
/DMP1	L2SCDMP1	
/DUMP	L2SCDUMP	
/DRCLASS	L2SCDRCL	
/DRMODE	L2SCDRMD	
/EADMIN	L2SCEADM	
/ECHO	N/A	
/EMAIL	L2SCEMAL	
/FETCH	N/A	
/GVAR	L2SCGVAR	
/IAS	L2SCIAS	
/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	
/OPERID	L2SCOPER	
/OPERIDS	L2SCOPRS	
/PA	N/A	
/PAGE	N/A	

Command	Resource Name	New in Version 12.0.00
/PF	N/A	
/PROF	N/A	
/PROFS	L2SCPROF	
/PURGPG	N/A	
/PURGPG(T)	L2SCPURG	
/REFRESH	L2SCRFSH	
/RELINK	L2SCRLNK	
/RESET	L2SCRSET	
/SDESK	L2SCSDSK	
/SHUTDOWN	L2SCSHUT	
/START	L2SCSTAR	
/STATEMGR	L2SCSTMG	
/STOP	L2SCSTOP	
/SWAP	L2SCSWAP	
/UID	L2SCUID	
/WLB	L2WBSWLB	
/WTO	L2SCWTO	
/XCF	L2SCXCF	
/XTASK	L2SCXTSK	
ADDRQ	L2QPADRQ	
ADDSCH	L2QPADSC	
AGFILE	L2AGX	
AGJOB	L2DB1	
AGPSWD	L2DBAPSW	
AL	L2UT1	
ALC	L2UT1C	
ALLOC	L2UT11	
ARFP	L2QPARFP	
ARTS	L2CAARTS	
BLDG	L2UT6	

Command	Resource Name	New in Version 12.0.00
CALMOD	L2DB28	
CANCEL	L2QPCNCL	
CAT	L2UT2	
CLEAR	N/A	
CONN	L2UT8	
CONVERT	L2DBCONV	
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	

Command	Resource Name	New in Version 12.0.00
FQALL	L2FCFQAL	
FQJOB	L2FCFQJO	
FQPOST	L2FCFQPO	
FQPRE	L2FCFQPR	
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	
JOBSTART	L2QPSTRT	
LACT	L2GILACT	

Command	Resource Name	New in Version 12.0.00
LACTR	L2GILACR	
LAGENT	L2GILAGT	
LARF	L2GILARF	
LARFQ	L2GILARQ	
LCTLG	L2GILCTL	
LDSN	L2GILDSN	
LDTM	L2GILDTM	
LGVAR	L2GILGVR	
LIST	L2GILIST	
LISTDIR	L2UT14	
LJCK	L2GILJCK	
LJCL	L2GILJCL	
LJES	L2GILJES	
LJOB	L2GILJOB	
LJOBR	L2GILJOR	
LLIB	L2GILLIB	
LLOCK	L2GILLOC	
LNODE	L2GILNOD	
LNTWK	L2GILNWK	
LOAD	L2QPLOAD	
LOADH	L2QPLOAD	
LOC	L2UT19	
LOGIN	L2QPLGIN	
LOGOUT	L2QPLGOU	
LPDS	L2GILPDS	
LPOST	L2GILPOS	
LPRE	L2GILPRE	
LPROS	L2GILPRO	
LPRRN	L2GILPRN	
LQ	L2GILQ	

Command	Resource Name	New in Version 12.0.00
LQP	L2GILQP	
LQUE	L2GILQ	
LQR	L2GILQR	
LRDY	L2GILRDY	
LRDYP	L2GILRDP	
LRDYR	L2GILRDR	
LREQ	L2GILREQ	
LREQP	L2GILREP	
LREQR	L2GILRER	
LRES	L2GILRES	
LRLOG	L2GILRLO	
LRMD	L2GILRMD	
LSCHD	L2GILSCH	
LSYS	L2GILSYS	
LTR	L2TSLTR	
LVAR	L2GILVAR	
LWLB	L2WBLWLB	
LXCF	L2GIXCF	
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	

Command	Resource Name	New in Version 12.0.00
PRRNDEL	L2QPPRND	
PRRNJCL	L2QPPRNJ	
PRSCF	L2QPPRCF	
PRSQA	L2QPPRQA	
PRSQD	L2QPPRQD	
PS	L2PS	
QJCL	L2QM5	
RELEASE	L2QPRLSE	
REMIND	L2QPRMIN	
RENAME	L2UT3	
REPLY	L2QPREPL	
REQUEUE	L2QPRQUE	
RESANL	L2DBRSNL	
RESCHNG	L2QPRSCH	
RESOLV	L2DBRSLV	
RESTART	L2QPREST	
RESTORE	L2DBCONV	
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	

Command	Resource Name	New in Version 12.0.00
START	L2QPSTAR	
STOP	L2QPSTOP	
SUBMIT	L2QPSUBM	
SUBSCH	L2QPSUBS	
SUBTM	L2QPSUBT	
SYSDMP	L2TSSYSD	
SYSINQ	L2TSSYSI	
TIQ	L2CATIQ	
TIQU	L2CATIQU	
TRA	L2TSTRA	
TRP	L2TSTRP	
UNC	L2UT5	
UT*	L2UT	
VERIFY	L2QPVERI	
X	L2TSX	
XNODE	L2DBXNOD	
XPJOB	L2DB1	
XPOST	L2QM7	
XPRE	L2QM6	
XPSWD	L2DBXPSW	
XQ	L2QM1	
XQJ	L2QM1	
XQM	L2QM1M	
XQN	L2QM1	
XRQ	L2QM2	
XRST	L2QM4	
XSPOST	L2QM7S	
XSPRE	L2QM6S	
XUPD	L2QM3	
XWLB	L2WBX	

Command	Resource Name	New in Version 12.0.00
ZA	L2TSZA	
ZAP	L2TSZAP	

Note: Restrict access to the commands with a resource name prefix of L2TS. These diagnostic commands are generally only issued at the request of CA Support. The ability to display and modify storage with these commands could allow unauthorized access of sensitive data.

Function and Service Level Table

The function table lists the CA WA CA 7 Edition 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 Access Level Translation Table (see page 171).

Functions	Service Level	Alias
ADD	ADD	A,ADDT,AELETE,AIST,APD
APPEND	READ	AP,APP
APPENDP	READ	N/A
CLEAR	N/A	CL,CLR
DD	DELETE	N/A
DELETE	DELETE	D,DEL,DELT
DELPRRN	UPDATE	N/A
EDIT	N/A	E,EDITH
EXIT	N/A	N/A
FE	READ	FEIT,FEPL,FEVE
FETCH	READ	F
FETCHP	READ	FP
FORMAT	N/A	FMT,FOR,FORM
FPE	READ	N/A
FREE	DELETE	N/A
LIST	READ	L,LDD,LDIT,LISTA,LISTP,LISTR,LPD

Functions	Service Level	Alias
PURGE	DELETE	N/A
RENAME	UPDATE	REN
REPL	UPDATE	R,REP,REPLH
REQ	UPDATE	N/A
RESOLV	SUBMIT	RES
RET	SUBMIT	N/A
RUN	SUBMIT	N/A
RUNH	SUBMIT	N/A
SAVE	ADD	S
SR	UPDATE	N/A
SS	ADD	N/A
SUBMIT	SUBMIT	SUB
UPD	UPDATE	U,UDD,UIST,UPDATE,UPDT
XPOST	UPDATE	N/A
XPRE	UPDATE	N/A
XQ	UPDATE	N/A
XQJ	UPDATE	N/A
XQM	UPDATE	N/A
XQN	UPDATE	N/A
XRQ	UPDATE	N/A
XRST	UPDATE	N/A
XSPOST	UPDATE	N/A
XSPRE	UPDATE	N/A
XUPD	UPDATE	N/A
XWLP	UPDATE	N/A

Access Level Translation Table

The following table describes the access levels for CA WA CA 7 Edition commands and panels and how they are translated by the CA Standard Security Facility (CAISSF). See 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 WA CA 7 Edition users.

CA WA CA 7 Edition	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

XML Query Resource Names

The following table is a list of XML query names that can be used through the Web Services facility of Jobflow Monitor. Each query contains a corresponding resource name to use when defining the resource rules to external security. All checks on these resources are for READ access.

XML Query Name	Resource Name
CancelSubscription	CA7JFM.XML.CANCELSUBSCRIPTION
CA7CmdRelease	CA7JFM.XML.CA7CMDRELEASE
CA7CmdHold	CA7JFM.XML.CA7CMDHOLD
CA7CmdCancel	CA7JFM.XML.CA7CMDCANCEL
CA7CmdDemand	CA7JFM.XML.CA7CMDDEMAND
CA7CmdRestart	CA7JFM.XML.CA7CMDRESTART
CA7CommandLineInterface	CA7JFM.XML.CA7COMMANDLINEINTERFACE
ExtractCa7Data	CA7JFM.XML.EXTRACTCA7DATA
GetCa7Data	CA7JFM.XML.GETCA7DATA
IssueConsoleMessage	CA7JFM.XML.ISSUECONSOLEMESSAGE
ListJobStatus	CA7JFM.XML.QUERYJOBSTATUS

XML Query Name	Resource Name
RetrieveCa7Data	CA7JFM.XML.RETRIEVECA7DATA
Subscription	CA7JFM.XML.SUBSCRIPTION