# Contents

**About this publication** ........................................ v  
- Intended audience ........................................... v  
- What this publication contains ................................. v  
- Access to publications and terminology ........................ vii  
- Related documentation ......................................... vii  
- Accessibility .................................................. ix  
- Technical training .............................................. ix  
- Support information ............................................ ix  
- Statement of Good Security Practices ........................... ix

**Chapter 1. Introduction** ..................................... 1  
- Application interface ......................................... 1  
- Command interface ............................................. 1  
- RRSF concerns .................................................. 2  
- Dates before and after the year 2000 .......................... 2

**Chapter 2. zSecure CICS Toolkit installation** ................. 5  
- Installation and post-installation checklist .................. 5  
  - Sample JCL .................................................... 6  
  - SMP/E zones creation and initialization ...................... 7  
  - Performing pre-installation steps ............................ 8  
  - TARGET and DLIB data sets allocation ....................... 8  
  - SMP/E DDDEFs update ....................................... 9  
  - Receive the product ......................................... 9  
  - zSecure CICS Toolkit code addition ........................ 9  
  - Integration of zSecure CICS Toolkit with your system .... 10  
  - Installing SVC .............................................. 10  
  - Protecting the SVC .......................................... 10  
  - Definition of SCQTLOAD as APF-authorized ................. 11  
  - Updating the CICS startup JCL .............................. 11  
  - zSecure CICS Toolkit enablement in PARMLIB .............. 11  
  - CQTPCNTL parameters definitions .......................... 12  
  - Definitions of programs, mapsets, and transactions to CICS 12  
  - Updating CICS tables ...................................... 13  
  - Defining the RACF profiles ................................ 14  
  - Internal Security Resource Listing ......................... 16  
  - Automatic assignment of USS UIDs (OMVS AUTOUID) ....... 19  
  - Automatic creation of home directories (OMVS MKDIR) ... 19  
  - zSecure CICS Toolkit restart ................................ 20  
  - RTST transaction definition ................................ 21  
  - Manual restart of the zSecure CICS Toolkit subtasks .... 21  
  - Use of the CICS Transaction Server with zSecure CICS Toolkit 22  
  - Globalization ............................................... 22

**Chapter 3. Parameters for zSecure CICS Toolkit** .......... 23  
- Parameter Descriptions ....................................... 23  
  - CQTPCNTL parameter values verification .................... 26

**Chapter 4. Application security management** ............... 29  
- Operator ID or OPID check .................................... 29  
- Application conversion ........................................ 29  
- Alias definitions .............................................. 31  
- Simple application security interface ........................ 31  
  - User information retrieval .................................. 31  
  - Resource access verification ................................. 32

**Chapter 5. The zSecure CICS Toolkit command interface** .... 35  
- Navigating the Main menu .................................... 35  
- Adding, altering, or deleting a group (ADDGROUP, ALTGROUPO or DELGROUP command) 36  
- Adding a user profile (ADDUSER command) .................. 37  
- Changing a profile (ALTUSER command) ....................... 39  
  - Altering the CICS segment for a user (ALTUSER CICS SEGMENT) 41  
  - Altering the TSO segment for a user (ALTUSER TSO SEGMENT) 43  
  - Altering the OMVS segment for a user (ALTUSER OMVS SEGMENT) 44  
  - Altering the WORKATTR segment for a user (ALTUSER WORKATTR SEGMENT) 47  
  - Connecting a user or group to a group (CONNECT command) 49  
  - Deleting a data set (DELETE DATASET command) .......... 50  
  - Deleting a user profile .................................... 50  
  - Listing the profile for one or more data sets (LISTDSET command) 51  
  - Toggling the LISTDSET panel ................................ 54  
  - Viewing the users authorized, their access authority and access count (LISTDSET USERIDS) 55  
  - Viewing the program/userid combination (LISTDSET Programs) 56  
  - Listing the profiles for one or more groups (LISTGROUP command) 57  
  - Toggling the LISTGROUP panel .............................. 60  
  - Listing users for a group (LISTGROUP command, USERIDS option) 60  
  - Deleting user IDs from a LISTGROUP ........................ 61  
  - Listing the subgroups of a group ............................ 62  
  - Viewing the users in a group (LISTUSER command) ........ 63  
  - Viewing the users in a group (LISTUSER command, Categories option) 68  
  - Viewing the users in a group (LISTUSER command, Categories option) 69

© Copyright IBM Corp. 1988, 2013
About this publication

The IBM® Security zSecure™ CICS® Toolkit enhances CICS/RACF security features by enabling you to execute RACF® commands directly from CICS and eliminating the need to use TSO. Application programs can also use IBM Security zSecure CICS Toolkit for security features instead of relying on internal application security functions. In this manner, all security definitions can be maintained centrally, or distributed among security coordinators.

This publication describes the two components of the IBM Security zSecure CICS Toolkit: the applications programming interface (API) and the command interface. It explains how to install and use this product.

Intended audience

This publication is intended for the following people:

- Systems support personnel responsible for the installation of IBM Security zSecure CICS Toolkit
- CICS Security administrators responsible for implementing the additional RACF command controls provided by IBM Security zSecure CICS Toolkit

Readers must also be familiar with performing security and administration tasks in a CICS environment and with RACF concepts and commands.

What this publication contains

This publication includes the following information:

- How to define parameters to be used by IBM Security zSecure CICS Toolkit. See Chapter 3, “Parameters for zSecure CICS Toolkit,” on page 23.
- How to use IBM Security zSecure CICS Toolkit to provide a central point for application security checks. See Chapter 4, “Application security management,” on page 29.
- How to define exit points to transfer control back to CICS after a IBM Security zSecure CICS Toolkit transaction completes. See Chapter 6, “zSecure CICS Toolkit exit points specifications,” on page 85.
- An API reference describing how to access the RACF database directly from a CICS application program. See Chapter 7, “Application programming interface (API),” on page 87.

For error messages, explanations, and workarounds where applicable, see IBM Security zSecure: Messages Guide.
Access to publications and terminology

This section provides:

- A list of publications in the "IBM Security zSecure library."
- Links to "Online publications" on page viii.
- A link to the "IBM Terminology website" on page viii.

IBM Security zSecure library

The following documents are available online in the IBM Security zSecure library:

- **IBM Security zSecure Release information**

- **IBM Security zSecure CARLa-Driven Components Installation and Deployment Guide, SC27-5638**
  Provides information about installing and configuring the following IBM Security zSecure components:
  - IBM Security zSecure Admin
  - IBM Security zSecure Audit for RACF, CA-ACF2, and CA-Top Secret
  - IBM Security zSecure Alert for RACF and ACF2
  - IBM Security zSecure Visual for RACF
  - IBM Tivoli® Compliance Insight Manager Enabler for z/OS®

- **IBM Security zSecure Admin and Audit for RACF Getting Started, GI13-2324**
  Provides a hands-on guide introducing IBM Security zSecure Admin and IBM Security zSecure Audit product features and user instructions for performing standard tasks and procedures. This manual is intended to help new users develop both a working knowledge of the basic IBM Security zSecure Admin and Audit for RACF system functionality and the ability to explore the other product features that are available.

- **IBM Security zSecure Admin and Audit for RACF User Reference Manual, LC27-5639**
  Describes the product features for IBM Security zSecure Admin and IBM Security zSecure Audit. Includes user instructions to run the features from ISPF panels, RACF administration and audit user documentation with both general and advanced user reference material for the CARLa command language and the SELECT/LIST fields. This manual also provides troubleshooting resources and instructions for installing the zSecure Collect for z/OS component. This publication is only available to licensed users.

- **IBM Security zSecure Audit for ACF2 Getting Started, GI13-2325**
  Describes the IBM Security zSecure Audit for ACF2 product features and provides user instructions for performing standard tasks and procedures such as analyzing Logon IDs, Rules, and Global System Options, and running reports. The manual also includes a list of common terms for those not familiar with ACF2 terminology.

- **IBM Security zSecure Audit for ACF2 User Reference Manual, LC27-5640**
  Explains how to use IBM Security zSecure Audit for ACF2 for mainframe security and monitoring. For new users, the guide provides an overview and conceptual information about using ACF2 and accessing functionality from the
ISPF panels. For advanced users, the manual provides detailed reference information including message and return code lists, troubleshooting tips, information about using zSecure Collect for z/OS, and details about user interface setup. This publication is only available to licensed users.

  Describes the IBM Security zSecure Audit for Top Secret product features and provides user instructions for performing standard tasks and procedures.

  Explains how to configure, use, and troubleshoot IBM Security zSecure Alert, a real-time monitor for z/OS systems protected with the Security Server (RACF) or CA-ACF2.

  Explains how to install and use IBM Security zSecure Command Verifier to protect RACF mainframe security by enforcing RACF policies as RACF commands are entered.

  Explains how to install and use IBM Security zSecure CICS Toolkit to provide RACF administration capabilities from the CICS environment.

- **IBM Security zSecure Messages Guide, SC27-5643**
  Provides a message reference for all IBM Security zSecure components. This guide describes the message types associated with each product or feature, and lists all IBM Security zSecure product messages and errors along with their severity levels sorted by message type. This guide also provides an explanation and any additional support information for each message.

- **IBM Security zSecure Quick Reference, SC27-5646**
  This booklet summarizes the commands and parameters for the following IBM Security zSecure Suite components: Admin, Audit, Alert, Collect, and Command Verifier. Obsolete commands are omitted.

  Explains how to set up and use the IBM Security zSecure Visual Client to perform RACF administrative tasks from the Windows-based GUI.

- **IBM Security zSecure Documentation CD, LCD7-5373**
  Supplies the IBM Security zSecure documentation, which contains the licensed and unlicensed product documentation. The IBM Security zSecure: Documentation CD is only available to licensed users.

- **Program Directory: IBM Security zSecure CARLa-Driven Components, GI13-2277**
  This program directory is intended for the system programmer responsible for program installation and maintenance. It contains information concerning the material and procedures associated with the installation of IBM Security zSecure CARLa-Driven Components: Admin, Audit, Visual, Alert, and the IBM Tivoli Compliance Insight Manager Enabler for z/OS. Program directories are provided with the product tapes. You can also download the latest copy from the IBM Security zSecure documentation website at [http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.zsecure.doc_2.1/welcome.html](http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.zsecure.doc_2.1/welcome.html).

- **Program Directory: IBM Security zSecure CICS Toolkit, GI13-2282**
  This program directory is intended for the system programmer responsible for program installation and maintenance. It contains information concerning the material and procedures associated with the installation of IBM Security zSecure CICS Toolkit. Program directories are provided with the product tapes. You can also download the latest copy from the IBM Security zSecure documentation
• **Program Directory: IBM Security zSecure Command Verifier, GI13-2284**
  This program directory is intended for the system programmer responsible for program installation and maintenance. It contains information concerning the material and procedures associated with the installation of IBM Security zSecure Command Verifier. Program directories are provided with the product tapes. You can also download the latest copy from the IBM Security zSecure documentation website at http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.zsecure.doc_2.1/welcome.html.

• **Program Directory: IBM Security zSecure Admin RACF-Offline, GI13-2278**
  This program directory is intended for the system programmer responsible for program installation and maintenance. It contains information concerning the material and procedures associated with the installation of the IBM Security zSecure Admin RACF-Offline component of IBM Security zSecure Admin. Program directories are provided with the product tapes. You can also download the latest copy from the IBM Security zSecure documentation website at http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.zsecure.doc_2.1/welcome.html.

**Online publications**

IBM posts product publications when the product is released and when the publications are updated at the following locations:

**IBM Security zSecure library**


**IBM Security Systems Documentation Central**

IBM Security Systems Documentation Central provides an alphabetical list of all IBM Security Systems product libraries and links to the online documentation for specific versions of each product.

**IBM Publications Center**

The IBM Publications Center site (http://www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss) offers customized search functions to help you find all the IBM publications you need.

**IBM Terminology website**


**Related documentation**

If you are using IBM Security zSecure Suite components in a RACF environment, you can find RACF user and reference information in several IBM manuals. The RACF commands and the implications of the various keywords can be found in the *RACF Command Language Reference* and the *RACF Security Administrator’s Guide*. Information about writing other RACF exits can be found in the *RACF System Programmer’s Guide*, while information about auditing RACF can be found in the *RACF Auditor’s Guide*. 

Table 1. Further information about RACF administration, auditing, programming, and commands

<table>
<thead>
<tr>
<th>Manual</th>
<th>Order Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS V1 Security Server RACF Command Language Reference</td>
<td>SA22-7687</td>
</tr>
<tr>
<td>z/OS V1 Security Server RACF System Administrator's Guide</td>
<td>SA22-7683</td>
</tr>
<tr>
<td>z/OS V1 Security Server RACF Auditor's Guide</td>
<td>SA22-7684</td>
</tr>
<tr>
<td>z/OS V1 Security Server RACF System Programmer's Guide</td>
<td>SA22-7681</td>
</tr>
<tr>
<td>z/OS MVS® System Commands</td>
<td>SA22-7627</td>
</tr>
</tbody>
</table>

Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully. With this product, you can use assistive technologies to hear and navigate the interface. You can also use the keyboard instead of the mouse to operate all features of the graphical user interface.

Technical training

For technical training information, see the following IBM Education website at http://www.ibm.com/software/tivoli/education.

Support information

IBM Support provides assistance with code-related problems and routine, short duration installation or usage questions. You can directly access the IBM Software Support site at http://www.ibm.com/software/support/probsub.html.

Statement of Good Security Practices

IT system security involves protecting systems and information through prevention, detection and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed, misappropriated or misused or can result in damage to or misuse of your systems, including for use in attacks on others. No IT system or product should be considered completely secure and no single product, service or security measure can be completely effective in preventing improper use or access. IBM systems, products and services are designed to be part of a comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective. IBM DOES NOT WARRANT THAT ANY SYSTEMS, PRODUCTS OR SERVICES ARE IMMUNE FROM, OR WILL MAKE YOUR ENTERPRISE IMMUNE FROM, THE MALICIOUS OR ILLEGAL CONDUCT OF ANY PARTY.
Chapter 1. Introduction

The zSecure CICS Toolkit enhances the CICS/RACF security features used by many installations. With zSecure CICS Toolkit, CICS Security Administrators can run RACF commands directly from CICS, eliminating the need to use TSO. Application programs can use zSecure CICS Toolkit for security features instead of having to rely on internal application security functions. In this manner, all security definitions can be maintained centrally, or distributed among security coordinators.

zSecure CICS Toolkit is divided into two parts: the application programming interface (API) and the command interface.

Applications can use zSecure CICS Toolkit for security instead of internal application security. You can maintain all security definitions centrally or distribute them among security coordinators.

Application interface

CICS systems programmers can use the zSecure CICS Toolkit API to customize the RACF commands and panels to their installation requirements.

In addition to having a central focal point for security definitions, the CICS systems programmer no longer needs to maintain the sign-on table; this table is no longer required for CICS version 2.x and later versions. As soon as transaction security checking is being performed by RACF, the only definition required in the sign-on table is the DEFAULT entry. Applications programmers no longer need to update their security tables and files because all the required information is defined to and maintained by RACF. Eliminating these tasks saves time that can be used for other work.

Another unique aspect of zSecure CICS Toolkit is the ability to RACF protect all application transactions. zSecure CICS Toolkit protects even those transactions that have not been protected previously. Examples of this protection are: a transaction that prints a report on a printer device, and an ATM transaction. Because printers and ATMs do not perform sign-ons, no security checking can be performed; therefore, transactions running on these devices could not be protected. zSecure CICS Toolkit removes this restriction so that all such transactions are protected. zSecure CICS Toolkit supports MRO environments and is easily installed, used, and maintained.

Command interface

CICS systems programmers can use the command interface to run certain RACF commands directly from CICS instead of TSO. The command interface provides greater distribution of certain capabilities and responsibilities to CICS users. It eliminates the need to use the TSO application and reduces CPU overhead.

When you use the command interface for searches:
- You can use more extensive search masks to retrieve information about profiles.
- You have greater flexibility in the criteria that are used.
Note: zSecure CICS Toolkit uses a different concept for enabling the user authority to run commands or portions of them. For example, with GROUPSPECIAL, a user without any more authority can reset a user password or resume the user. This user does not require GROUPSPECIAL and does not have to be connected to the group of the user that is being reset.

Under this methodology, the responsibility of resetting user passwords can be decentralized to any other area where there is access to a CICS terminal. Data Security personnel can then use their time and efforts in other areas.

RACF protects both the transaction that is used to start zSecure CICS Toolkit for the command interface and the commands themselves. Even if there is no security on the zSecure CICS Toolkit transaction, the user must be permitted to the commands in RACF to start them. A user must also be permitted to have the authority to reset user IDs. The only exception is a user with the SPECIAL attribute or a user with access to the TOOLKIT.SPEC definition. For information about this definition, see Chapter 2, “zSecure CICS Toolkit installation,” on page 5.

All changes to RACF profiles produce SMF records.

RRSF concerns

There is a difference between the zSecure CICS Toolkit actions and the TSO RACF commands.

All additions, updates, and deletions of RACF profiles that zSecure CICS Toolkit does are done with either the RACROUTE or ICHEINTY interface. In the remainder of this manual, the function is called by the equivalent RACF command processor name. For example, when the manual talks about the API interface that connects a user to a group, it is called the CONNECT command. The actual processing is not the CONNECT command. It is a similar set of actions that cause the same result as if the RACF CONNECT command was used. Therefore, the updates are considered application updates. The difference between the TSO RACF command and the zSecure CICS Toolkit actions is only apparent in the required RACF Remote Sharing Facility (RRSF) definitions if you need RRSF propagation of updates.

If you want to propagate changes through RRSF, activate this function by the operator RACF SET AUTOAPPL command either from the console, or by PARMLIB. You probably have already set the equivalent SET AUTODIRECT option to propagate RACF commands. For the automatic propagation of application updates, you must also define appropriate profiles in the RRSFDATA resource class. The profiles must allow the RACF updates that the CICS region makes to be propagated. The user that needs access to the RRSFDATA profiles is the CICS region user, and not the CICS terminal user or the CICS default user. For more information about the definition of appropriate RRSFDATA profiles, see the "Automatic Direction" section in the RACF Remote Sharing Facility chapter in the RACF Security Administrator's Guide.

Dates before and after the year 2000

When a date is specified for the REVOKEDATE and RESUMEDATE parameters, zSecure CICS Toolkit treats the dates according to IBM rules.

The date format is YYDDD, where YY is the year and DDD is the day number.
• If the specification for YY is 71 or greater, the date is treated as being in the 20th century; for example, 1971.
• If the specification for YY is 70 or less, the date is treated as being in the 21st century; for example, 2070.

These rules also provide compatibility with previous releases.
Chapter 2. zSecure CICS Toolkit installation

Systems support personnel can use the information in this chapter to install zSecure CICS Toolkit. SMP/E is used for the installation process.

Before you begin the installation, see the Program Directory: IBM Security zSecure CICS Toolkit for information about the prerequisites. Then, see “Installation and post-installation checklist” for pointers to instructions for installation and post-installation tasks.

### Installation and post-installation checklist

Systems support personnel responsible for installing zSecure CICS Toolkit can use the checklist below to perform installation and post-installation tasks.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Instructions</th>
<th>Job name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Load installation JCL.</td>
<td>“Sample JCL” on page 6</td>
<td>CQTJSMPx (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>2a</td>
<td>Create and initialize SMP/E zones.</td>
<td>“SMP/E zones creation and initialization” on page 7</td>
<td>CQTJALL (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>2b</td>
<td>Allocate TARGET and DLIB data sets.</td>
<td>“TARGET and DLIB data sets allocation” on page 8</td>
<td>CQTJALL (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Update SMP/E DDEFs.</td>
<td>“SMP/E DDDEFs update” on page 9</td>
<td>CQTJDDD (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>4a</td>
<td>RECEIVE zSecure CICS Toolkit.</td>
<td>“Receive the product” on page 9</td>
<td>CQTJREC (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>4b</td>
<td>APPLY zSecure CICS Toolkit.</td>
<td>“zSecure CICS Toolkit code addition” on page 9</td>
<td>CQTJAPP (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>4c</td>
<td>ACCEPT zSecure CICS Toolkit.</td>
<td>“Integration of zSecure CICS Toolkit with your system” on page 9</td>
<td>CQTJACC (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Install the SVC.</td>
<td>“Installing SVC” on page 10</td>
<td>CQTJDEF (SCQTSAMP)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Protect the SVC.</td>
<td>“Protecting the SVC” on page 10</td>
<td>CQTJDEF (SCQTSAMP)</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Define data sets as APF-authorized.</td>
<td>“Definition of SCQTLOAD as APF-authorized” on page 11</td>
<td>CQTJDEF (SCQTSAMP)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Update the CICS startup JCL.</td>
<td>“Updating the CICS startup JCL” on page 11</td>
<td>CQTJDEF (SCQTSAMP)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Check product enablement through IFAPRDxx in PARMLIB.</td>
<td>“zSecure CICS Toolkit enablement in PARMLIB” on page 11</td>
<td>CQTJCNTL (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Update the CQTPCNTL parameters.</td>
<td>“CQTPCNTL parameters definitions” on page 12</td>
<td>CQTJCNTL (SCQTINST)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Define programs, mapsets, and transactions to CICS.</td>
<td>“Definitions of programs, mapsets, and transactions to CICS” on page 12</td>
<td>CQTJRDO (SCQTSAMP)</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Update the CICS tables.</td>
<td>“Updating CICS tables” on page 13</td>
<td>CQTJPLT CQTJSRT (SCQTSAMP)</td>
<td></td>
</tr>
</tbody>
</table>
Table 2. Installation checklist (continued)

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Instructions</th>
<th>Job name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Define RACF profiles to control access to zSecure CICS Toolkit functions.</td>
<td>&quot;Defining the RACF profiles&quot; on page 14</td>
<td>CQTJRDEF (SCQTSAMP)</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Grant more authorizations to the CICS Started Task to perform USS functions.</td>
<td>&quot;Automatic assignment of USS UIDs (OMVS AUTOUID)&quot; on page 19 and &quot;Automatic creation of home directories (OMVS MKDIR)&quot; on page 19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Restart zSecure CICS Toolkit.</td>
<td>&quot;zSecure CICS Toolkit restart&quot; on page 20</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sample JCL

Systems support personnel responsible for installing zSecure CICS Toolkit can access the installation jobs from relfile 2 and the SCQTSAMP data set or copy the jobs from the tape or product files.

To access the sample installation jobs, perform an SMP/E RECEIVE. Then copy the jobs from the relfiles to a work data set for editing and submission. The sample JCL used for the SMP/E steps is contained in relfile 2. The remaining sample JCL, used during the configuration steps, is contained in the SCQTSAMP data set that is created during the earlier installation steps.

To copy the jobs from the tape or product files, submit the job that follows this paragraph. Use either the //TAPEIN or the //FILEIN DD statement, depending on your distribution medium, and comment out or delete the other statement. Add a job card and change the lowercase parameters to uppercase values to meet your site requirements before you submit.

```plaintext
//STEP1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=* //TAPEIN DD DSN=IBM.HCQT1D1.F2,UNIT=tunit, //  VOL=SER=volser, LABEL=(x,SL), //  DISP=(OLD,KEEP) //FILEIN DD DSN=IBM.HCQT1D1.F2,UNIT=SYSALLDA,DISP=SHR, //  VOL=SER=filevol //OUT DD DSNNAME=jcl-library-name, //  DISP=(NEW,CATLG,DELETE), //  VOL=SER=dasvol,UNIT=SYSALLDA, //  SPACE=(TRK,(5,5,5)) //SYSUT3 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1)) //SYSIN DD * COPY INDD=xxxxIN,OUTDD=OUT /*
```

In the previous sample, update the statements as follows:

- If you use TAPEIN:
  - `tunit` The unit value that matches the product tape.
  - `volser` The volume serial that matches the product tape.
  - `x` The tape file number where the data set name is on the tape.

See the documentation that is provided by CBPDO to see where IBM.HCQT1D0.F2 is on the tape.
If you install from a non-CBPDO product tape, the tape volser is CQT1D0, and the file number for relfile 2 is 3 on the tape.

- If you use FILEIN:
  
  `filevol`  The volume serial of the DASD device where the downloaded files is located.

- In the OUT statement:
  
  `jcl-library-name`  The name of the output data set where the sample jobs are stored.

  `dasdvol`  The volume serial of the DASD device where the output data set is located.

- In the SYSIN statement:
  
  `xxxIIN`  Either TAPEIN or FILEIN, depending on your input DD statement.

### SMP/E zones creation and initialization

Before starting the SMP/E installation, systems support personnel responsible for installing zSecure CICS Toolkit must decide on the SMP/E zones to use.

You can choose one of the following options:

- Install zSecure CICS Toolkit in new (dedicated) zones in a new CSI.  
  This option is the only one for which sample jobs are provided.
- Install zSecure CICS Toolkit in new (dedicated) zones in an existing CSI.
- Install zSecure CICS Toolkit in existing zones in an existing CSI.

The sample jobs that are provided do not accommodate all possible combinations of CSIs and zones. The only option for which sample jobs are provided is the first option. The jobs can be used to set up a dedicated GLOBAL and PRODUCT CSI with dedicated TARGET and DLIB zones. For information, see "Performing pre-installation steps" on page 8.

The example jobs are provided in SCQTINST. Adapt the JCL and submit the jobs to complete the creation and initialization of the SMP/E environment. The jobs all use lowercase strings for the values that must be adapted to fit your installation standards. The following values are currently used.

**Your-Global**

The data set prefix that you want to use for the GLOBAL SMP/E data sets. This prefix is used for the name of the GLOBAL CSI and for the SMP/E data sets shared between all SMP/E zones.

**Your-Product**

The data set prefix that you want to use for the zSecure CICS Toolkit data sets. This data set prefix is also the prefix for the SMP/E data sets specific to zSecure CICS Toolkit.

**sysallda**

The unit name that is used for all data set allocations.

**volser**

The name of the DASD volume in your system where you want to create the zSecure CICS Toolkit data sets.

**Note:** In an SMS environment, the ACS routines might assign another volume than the one specified from the `volser`.  

Chapter 2. zSecure CICS Toolkit installation 7
tape The unit name of the tape-unit where the zSecure CICS Toolkit distribution tape can be mounted.

Note: The value for Your-Global cannot be the same as that for Your-Product. If you want to use similar prefixes, add an extra qualifier for the GLOBAL zone. For example, you might use the following two values:

- SMPE.TOOOLKIT.GLOBAL as the value for Your-Global
- SMPE.TOOOLKIT as the value for Your-Product

Use the following table to record the values for the installation variables suitable for your environment.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Your Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Your-Global</td>
<td></td>
</tr>
<tr>
<td>Your-Product</td>
<td></td>
</tr>
<tr>
<td>sysallda</td>
<td></td>
</tr>
<tr>
<td>volser</td>
<td></td>
</tr>
<tr>
<td>tape</td>
<td></td>
</tr>
</tbody>
</table>

Performing pre-installation steps

Before beginning the installation, systems support personnel responsible for installing zSecure CICS Toolkit must define the required SMP/E zones.

Procedure

1. Create a global CSI: The sample job for creation of a new GLOBAL CSI is provided in member CQTJSMPA. This job also defines a GLOBAL zone in this new CSI.
   - Adapt and submit job CQTJSMPA.
2. Create a product CSI: The sample job for creation of a new product CSI is provided in member CQTJSMPB. This job also defines a TARGET and DLIB zone in this new CSI.
   - Adapt and submit job CQTJSMPB.
3. Define an options entry for zSecure CICS Toolkit: The sample job for defining an options entry specific for zSecure CICS Toolkit is provided in member CQTJSMPC. It specifies the utilities and the data set prefix to be used during the remaining SMP/E installation steps.
   - Adapt and submit job CQTJSMPC.

TARGET and DLIB data sets allocation

Systems support personnel responsible for installing zSecure CICS Toolkit must allocate the TARGET and DLIB data sets.

zSecure CICS Toolkit adds six target data sets and six distribution data sets to your SMP/E environment. Example JOB CQTJALL contains the necessary JCL to allocate the required TARGET and DLIB data sets.

- Submit CQTJALL
**SMP/E DDDEFs update**

Systems support personnel responsible for installing zSecure CICS Toolkit perform this step to define to SMP/E the data sets that you allocated in the previous step. If you choose to include appropriate DD-statements in all of your SMP/E jobs, you can omit this step. If you want to use the suggested setup through dynamic allocation, this step is required. The example job CQTJDDD contains the JCL needed for this step.

The installation process also uses SMP/E CALLLIBS processing. This function is used to resolve external references during installation. When zSecure CICS Toolkit is installed, ensure that DDDEFs exist for the CSSLIB library.

**Note:** The DDDEFs are used only to resolve the link edit for zSecure CICS Toolkit with CALLLIBS. These data sets are not updated during the installation of zSecure CICS Toolkit. The provided sample job includes the required DDDEF.

» Submit job CQTJDDD.

**Receive the product**

Systems support personnel need the SMP/E Modification Control Statements to correctly install zSecure CICS Toolkit.

If you are installing from the zSecure CICS Toolkit product tape, the first file on that tape is the SMPMCS data set. It contains the SMP/E Modification Control Statements that are needed for correct installation of zSecure CICS Toolkit.

» Submit job CQTJREC.

**zSecure CICS Toolkit code addition**

Systems support personnel responsible for installing zSecure CICS Toolkit must add the code, examples, and documentation to the system.

The SMP/E statement that is needed is

```
APPLY GROUPEXTEND SELECT(HCQT1D1).
```

Because of the use of a SELECT for the product FMID, SMP/E does not require the use of the FUNCTIONS keyword. An example job is included in member CQTJAPP. Before you run this job, specify the data set name of your GLOBAL CSI.

» Submit the job CQTJAPP.

**Integration of zSecure CICS Toolkit with your system**

If systems support personnel responsible for the installation of zSecure CICS Toolkit are satisfied with its implementation, they can ACCEPT the product. After you ACCEPT, the product is a part of your system.

zSecure CICS Toolkit does not normally cause any further systems programming work. An example ACCEPT job is provided in CQTJACC.

» Submit job CQTJACC.
Installing SVC

Systems support personnel must install a type 3 SVC to be used for zSecure CICS Toolkit.

Procedure

1. Include the SCQTLPA data set in your LPALSTxx member in PARMLIB. You can also copy SCQTSVC00 from the supplied SCQTLPA to an existing LPALIST data set. For example, SYS1.LPALIB.
2. Determine the SVC number to be used for the zSecure CICS Toolkit SVC. In this example, use 222.
3. Perform the following steps:
   a. Update the IEASVCCxx member in PARMLIB with an entry to define the zSecure CICS Toolkit SVC. An example is
      ```
      SVC>PARM 222,REPLACE,TYPE(3),APF(NO),EPNAME(CQTSVC00)
      ```
   b. Perform an IPL to install the SVC into the system.

   Note: Some software packages provide the capability of updating the SVC table without performing an IPL. If you are using such a software package, you might not need an IPL to start using the SVC.
4. Add the DFHSRT entry with the following format:
   ```
   DFHSRT TYPE=SYSTEM,ABCODE=XX,ROUTINE=DFHSRTRR
   ```
   where XX is the hexadecimal number of the SVC. For example, if the number assigned to the SVC is 222, the entry is FDE. For more information about updating the DFHSRT table, see "Updating CICS tables" on page 13.

Protecting the SVC

To prevent unauthorized use of the zSecure CICS Toolkit SVC, a RACHECK is issued from the SVC to ensure that the caller is authorized.

About this task

Before CICS can use zSecure CICS Toolkit, it must be given access to through RACF.

Procedure

1. Define the SVC to RACF with the DEFINE command. You must use resource class FACILITY.
   ```
   RDEFINE FACILITY TOOLKIT.SVC UACC(NONE)
   ```
2. Grant SVC access to each CICS region that will have an installation of the toolkit. Use the following command:
   ```
   PERMIT TOOLKIT.SVC CLASS(FACILITY) ID(userid) ACCESS(READ)
   ```
   where userid is the ID of the CICS region.
   A sample of the preceding RACF definitions is included in member CQTJREDF in SCQTSAMP.
**Definition of SCQTLOAD as APF-authorized**

The zSecure CICS Toolkit subtask programs perform the retrieval and updates to the profiles in the RACF database. Use of these RACF functions requires APF authorization.

The data set containing these subtask programs is SCQTLOAD. This data set must be APF-authorized. To define SCQTLOAD as APF-authorized, you must update the IEAAPFxx or PROGxx member in PARMLIB with the name of the SCQTLOAD data set.

> Update PROGxx and activate by operator command SET PROD=xx.

**Updating the CICS startup JCL**

Several changes must be made to the CICS startup JCL.

**Procedure**

1. Add the SCQTRPL data set that contains the zSecure CICS Toolkit programs and maps to the DFHRPL concatenation. For example:
   ```
   //DFHRPL DD DISP=SHR,DSN=APPL1.LOADLIB
   //   DD DISP=SHR,DSN=APPL2.LOADLIB
   //   DD DISP=SHR,DSN=APPL3.LOADLIB
   //   DD DISP=SHR,DSN=APPL4.LOADLIB
   //   DD DISP=SHR,DSN=CICS.TOOLKIT.SCQTRPL
   ```

2. The SCQTLOAD data set contains the zSecure CICS Toolkit programs that are used as MVS subtasks. Add this data set to the CICS STEPLIB.

3. If you decide to add a zSecure CICS Toolkit loadlib to the STEPLIB concatenation, rather than copying the modules to an existing loadlib, you must also make that loadlib APF-authorized because all loadlibs in the CICS STEPLIB must be APF-authorized.

   To define the zSecure CICS Toolkit loadlib as APF-authorized, you must update the IEAAPFxx or PROGxx member in SYS1.PARMLIB with the name of the zSecure CICS Toolkit loadlib. An example of updating the JCL STEPLIB parameter:
   ```
   //STEPLIB DD DISP=SHR,DSN=LOADLIB1
   //   DD DISP=SHR,DSN=LOADLIB2
   //   DD DISP=SHR,DSN=CICS.TOOLKIT.SCQTLOAD
   ```

**zSecure CICS Toolkit enablement in PARMLIB**

At startup of the zSecure CICS Toolkit tasks, zSecure CICS Toolkit verifies whether the product is enabled or disabled by IFAPRDxx in PARMLIB.

If the product is enabled or not defined in IFAPRDxx, initialization of zSecure CICS Toolkit continues normally.

If the product is disabled, a message (CQT907) is issued, and zSecure CICS Toolkit initialization is terminated. Disabling the product does not affect the remainder of CICS initialization.

To explicitly enable zSecure CICS Toolkit, include an entry such as the following one in an active IFAPRDxx member.

```
PRODUCT OWNER('IBM CORP')
   NAME('zSecure Toolkit')
   ID(5655-N18)
   VERSION(*) RELEASE(*) MOD(*)
   STATE(ENABLED)
```
If you want to disable zSecure CICS Toolkit, create an entry like this one, and replace the parameter \texttt{STATE(ENABLED)} by the parameter \texttt{STATE(DISABLED)}.

- Update IFAPRDxx and activate by operator command \texttt{SET PROD=xx}.

**CQTPCNTL parameters definitions**

CQTPCNTL defines certain parameters that zSecure CICS Toolkit uses.

These parameters include the SVC number and the RACF resource class that is used. See Chapter 3, “Parameters for zSecure CICS Toolkit,” on page 23 for CQTPCNTL parameter definitions and information about customizing CQTPCNTL.

- Adapt CQTPCNTL and Submit CQTJCNTL.

After the definitions are made, transaction RTCK can be run to check the parameters in CQTPCNTL.

**Definitions of programs, mapsets, and transactions to CICS**

You must define the programs, mapsets, and transactions that zSecure CICS Toolkit uses. Although you can still use CICS tables for this definition, the preferred method is through CICS Resource Definition Online (RDO).

An example job to make these definitions is provided in member CQTJRD0 in SCQTSAMP. You must define the following mapsets:

- CQTBST0
- CQTBCH0
- CQTB000
- CQTB100
- CQTB200
- CQTB300
- CQTB400
- CQTB500
- CQTB550
- CQTB560
- CQTB580
- CQTB590
- CQTB600
- CQTB700
- CQTB800
- CQTB860
- CQTB900
- CQTBA00
- CQTB000
- CQTB000
- CQTB000

You must define the following programs:

- CQTPAPI0
- CQTPSNP0
- CQTPATCH
- CQTPCHECK
- CQTPDTC0
- CQTPAPI0
- CQTP040
- CQTP041
- CQTP042
- CQTP043
- CQTP044
- CQTP050
- CQTP055
- CQTP056
- CQTP057
- CQTP058
- CQTP059
- CQTP060
- CQTP070
- CQTP080
- CQTP081
- CQTP082
- CQTP083
- CQTP084
- CQTP086
- CQTP090
- CQTP091
- CQTP100
- CQTP110
- CQTP111
- CQTP112
- CQTP113
- CQTP114
- CQTP120
- CQTP130
- CQTP131
- CQTP132
- CQTP133
- CQTP134
- CQTP135
- CQTP136
- CQTP140
- CQTPAPPL

In addition to the regular executable programs, some modules contain data that is used by all zSecure CICS Toolkit programs. This data must be permanently available to all zSecure CICS Toolkit programs. You can make the data permanently available by defining these programs as \textit{resident}. The following programs must be defined as resident programs:

- CQTPAPRM
- CQTPMSGE
- CQTPCNC1

All the zSecure CICS Toolkit programs (both the regular programs and the resident programs) must be defined with \texttt{EXECKEY(CICS)}.

You can define the following programs for compatibility reasons. Your application programs can refer to these programs by the name that is used in the precursor product Consul zToolkit. These programs were renamed in Tivoli zSecure CICS Toolkit version 1.8.1. For best results, use the following names:
If you did not adapt your existing applications to reflect the new names, you must also define the following programs. These programs are alias names of the new modules:

```
CRTKAPI  CRTKSNP  CRTKAPPL
```

To use the required functions of zSecure CICS Toolkit, you might also need to define the online transactions. If you use only the functions from the API interface, definition of these transactions is not required. The following transaction must be defined for the following programs:

```
RTCK --> PROG(CQTPCHEK)
RTST --> PROG(CQTPSTRT)
RTMM --> PROG(CQTP0000)
```

- Update and submit CQTJRDO.

If you want these definitions to be active at startup, you must also include the `group` TOOLKIT in a list that is used to activate CICS resources.

### Updating CICS tables

Systems support personnel must update the CICS tables.

#### Procedure

- Add the following entry to DFHPLTPI:

  ```
  DFHPLT TYPE=ENTRY,PROGRAM=CQTPLT00
  ```

  **Note:** CQTPLT00 must be placed after the DFHDELIM entry in the PLT.

- Add the following entry to DFHPLTSD:

  ```
  DFHPLT TYPE=ENTRY,PROGRAM=CQTPDTCH
  ```

  Without the shutdown entry in the PLT, CICS is likely to abend with system abend code A03 during regular shutdown processing.

- Add the following entry to DFHSRT:

  ```
  DFHSRT TYPE=SYSTEM,ABCODE=Fxx,RECOVER=YES
  ```

  For the exact definition of the parameter ABCODE=Fxx, see "Installing SVC" on page 10.

  The previous tables must be translated with the CICS table update procedure in use at your installation. Typically, the procedure is called DFHAULPE, and it is in a data set named similarly to CICSTS31.XDFHINST. An example is provided in job CQTJPLT and CQTJSRT.

- Adapt and submit job CQTJPLT.
- Adapt and submit CQTJSRT.

  **Note:** The PLT program must be found during CICS startup. That means that the CQTPLT00 program must already be defined to CICS. The RDO example defines the resources, but does not automatically activate them. You must either include the TOOLKIT `group` in a list, or ensure that the resource definition is active in some other way.
Defining the RACF profiles

Follow these steps to use the zSecure CICS Toolkit RACF command interface to define RACF profiles.

Procedure

1. Define the zSecure CICS Toolkit commands to RACF.

Table 4 contains a list of the zSecure CICS Toolkit commands and the first and subsequent levels of authority that is required for each:

Table 4. zSecure CICS Toolkit commands: Required authorization levels

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>LEVEL</th>
<th>SUBSEQUENT LEVELS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDGROUP</td>
<td>TOOLKIT.ADGR</td>
<td>ADGR.grpname</td>
</tr>
<tr>
<td>ADDUSER</td>
<td>TOOLKIT.ADUS</td>
<td>ADUS.dfltgrp</td>
</tr>
<tr>
<td>ALTGROUP</td>
<td>TOOLKIT.ALGR</td>
<td>ALGR.grpname</td>
</tr>
<tr>
<td>ALTUSER</td>
<td>TOOLKIT.AUSR</td>
<td>AUSR.dfltgrp When you assign a shared UID in an OMVS segment, you also need either System-SPECIAL or access to SHARED.IDS in the UNIXPRIV class.</td>
</tr>
<tr>
<td>CONNECT</td>
<td>TOOLKIT.CONN</td>
<td>CONN.grpname</td>
</tr>
<tr>
<td>DELDSD</td>
<td>TOOLKIT.DELD</td>
<td>DELD.hlq</td>
</tr>
<tr>
<td>DELGROUP</td>
<td>TOOLKIT.DELG</td>
<td>DELG.grpname</td>
</tr>
<tr>
<td>DELUSER</td>
<td>TOOLKIT.DELU</td>
<td>DELU.dfltgrp</td>
</tr>
<tr>
<td>LISTDATASET</td>
<td>TOOLKIT.LDSD</td>
<td>None</td>
</tr>
<tr>
<td>LISTGROUP</td>
<td>TOOLKIT.LGRP</td>
<td>LGRP.grpname</td>
</tr>
<tr>
<td>LISTUSER</td>
<td>TOOLKIT.LUSR</td>
<td>LUSR.dfltgrp</td>
</tr>
<tr>
<td>PASSWORD</td>
<td>None</td>
<td>PSWD.grpname</td>
</tr>
</tbody>
</table>
The PASSWORD command is available only through the API. |
<p>| PERMIT      | TOOLKIT.PEMT | PEMT.grpname / grpname                   |
|             |           | If the PERMIT is for a GROUP, PEMT.grpname is used. If it is for a USERID, then PEMT.grpname is used. |
|             |           | When issuing a PERMIT, the user also requires access to the resource that is being given access to. |
|             |           | If the resource is in a class that is not defined in the CICS SIT, the user also needs access to PEMX.cdtclass. |
| RACLK       | TOOLKIT.RACL | RACL.dfltgrp                             |
| RALTER      | TOOLKIT.RALT | RALT.cdtclass                           |
| RDEFINE     | TOOLKIT.RDEF | RDEF.cdtclass                           |
| REMOVE      | TOOLKIT.REMV | REMV.grpname                           |</p>
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>LEVEL</th>
<th>SUBSEQUENT LEVELS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDELETE</td>
<td>TOOLKIT.RDEL</td>
<td>RDEL.cdtclass</td>
</tr>
<tr>
<td>RLIST</td>
<td>TOOLKIT.RLST</td>
<td>RLST.cdtclass</td>
</tr>
<tr>
<td>USRDATA</td>
<td>TOOLKIT.USRL</td>
<td>USRU.grpname USRN.usrdata-name</td>
</tr>
</tbody>
</table>

When using the subfunctions to Add/Update/Delete, or when accessing these functions directly from the API, access to TOOLKIT.USRA or TOOLKIT.USRD is required.

<table>
<thead>
<tr>
<th>VERIFY</th>
<th>None</th>
</tr>
</thead>
<tbody>
<tr>
<td>VERIFY is available only through the API and allows applications to verify a user’s ID and password without the need for the user to sign on.</td>
<td>None</td>
</tr>
</tbody>
</table>

In all of the preceding definitions:
- `grpname` is the GROUP name.
- `dfltgrp` is the DEFAULT GROUP name.
- `hlq` is the high-level-qualifier of the data set name.
- `cdtclass` is the GENERAL RESOURCE CLASS name that is defined in the CDT.

Each command is defined to RACF as a resource in a manner similar to the one in which transactions are defined.

It is best to define the following generic names first, with a UACC of NONE. This ensures that users are not given access to commands through other generic definitions within the resource class.

Assuming that the RSRCLASS parameter in CQTPCNTL is TCICSTRN:

```r
RDEFINE TCICSTRN (TOOLKIT.* ADGR.* ADUS.* ALGR.* AUSR.*
CONN.* DELD.* DELG.* DELU.* LDSD.* LGRP.* LUSR.* PEMT.*
PSWD.* RACL.* RALT.* RDEF.* RDEL.* REMV.* RLST.*
USRU.* USRN.*) UACC(NONE)
```

Define the zSecure CICS Toolkit commands to RACF with the RDEFINE command. This example also assumes that TCICSTRN is the RSRCLASS parameter.

```r
RDEFINE TCICSTRN (TOOLKIT.ADGR TOOLKIT.ADUS TOOLKIT.ALGR TOOLKIT.AUSR
TOOLKIT.CONN TOOLKIT.DELD TOOLKIT.DELG TOOLKIT.DELU TOOLKIT.LDSD TOOLKIT.LGRP
TOOLKIT.LUSR TOOLKIT.PEMT TOOLKIT.REMV TOOLKIT.RACL TOOLKIT.RALT TOOLKIT.RDEF
TOOLKIT.RDEL TOOLKIT.RLST TOOLKIT.USRL TOOLKIT.USRA TOOLKIT.USRD)
```

2. Give the users access to the commands that they can be authorized to use. For example, ADDUSER, ALTUSER, DELUSER.

When you use the software, use the authority of the user signed on at the terminal to determine which commands and functions can be run. If a transaction that is not running at a terminal uses the software to perform a RACF function, the authority of the DEFAULT USERID is used to determine which commands and functions can be executed. The authority of the default user is also used for terminals that are not explicitly signed-on.

Permit users to each command as required. Examples:
Only commands that a user has access to are displayed on the panel. For example, if a user has access to only the LUSR command, none of the other commands are displayed.

3. Define the subsequent levels of authority within those commands that the user can access. For example, after you give a user access to the ALTUSER command, you must specify which users they can alter.

**Note:** For a list of internal security resource classes for the commands, see “Internal Security Resource Listing.”

Most commands require one or more subsequent levels of authority to be given to a user. These next levels determine which users, groups, and resources the user can access or control after you have given the user access to a command. For example, if a user is granted access to ALTUSER, the IDs to which the user has access must also be defined. This is done by defining the default group of users, prefixed with AUSR, as a resource. A user can then be granted access to this resource.

Table 5 shows examples of definitions that are required to give this type of capability.

**Table 5. Example resource definitions used within RACF commands**

<table>
<thead>
<tr>
<th>USERID</th>
<th>DEFAULT GROUP</th>
<th>RSRCLASS</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER01</td>
<td>TECHSUPP</td>
<td>TCICSTRN</td>
</tr>
<tr>
<td>USER02</td>
<td>USERSUPP</td>
<td>TCICSTRN</td>
</tr>
<tr>
<td>USER03</td>
<td>QUALCNTL</td>
<td>TCICSTRN</td>
</tr>
<tr>
<td>USER04</td>
<td>AUDIT</td>
<td>TCICSTRN</td>
</tr>
</tbody>
</table>

The RACF command in this example:

```
RDEFINE TCICSTRN (AUSR.TECHSUPP AUSR.USERSUPP AUSR.QUALCNTL AUSR.AUDIT)
```

Then, permit users to each group:

```
PERMIT AUSR.TECHSUPP CLASS(TCICSTRN) ID(USER01) ACCESS(READ)
PERMIT AUSR.QUALCNTL CLASS(TCICSTRN) ID(USER01) ACCESS(READ)
PERMIT AUSR.AUDIT CLASS(TCICSTRN) ID(USER01) ACCESS(READ)
```

With these definitions, USER01 can perform an ALTUSER on all user IDs whose default group is TECHSUPP, QUALCNTL, or AUDIT, but not on user IDs whose default group is USERSUPP, or any other default group.

**Internal Security Resource Listing**

zSecure CICS Toolkit uses resource names for its own internal security.

Table 6 lists the resource names that zSecure CICS Toolkit uses for its own internal security.

**Table 6. zSecure CICS Toolkit: Internal security resource list**

<table>
<thead>
<tr>
<th>Resource name</th>
<th>What it allows if a user has read access</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOOLKIT.ADGR</td>
<td>Allows a user to run the ADDGROUP command of zSecure CICS Toolkit.</td>
</tr>
</tbody>
</table>
Table 6. zSecure CICS Toolkit: Internal security resource list (continued)

<table>
<thead>
<tr>
<th>Resource name</th>
<th>What it allows if a user has read access</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOOLKIT.ADUS</td>
<td>Allows a user to run the ADDUSER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.ALGR</td>
<td>Allows a user to run the ALTGROUP command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.AUSR</td>
<td>Allows a user to run the ALTUSER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.ACIC</td>
<td>Allows a user to manage CICS segments by the ALTUSER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.ATSO</td>
<td>Allows a user to manage TSO segments by the ALTUSER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.AOMV</td>
<td>Allows a user to manage OMVS segments by the ALTUSER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.AWRK</td>
<td>Allows a user to manage WORKATTR segments by the ALTUSER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.CONN</td>
<td>Allows a user to run the CONNECT command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.DELD</td>
<td>Allows a user to run the DELETE DATASET command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.DELG</td>
<td>Allows a user to run the DELETE GROUP command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.DELU</td>
<td>Allows a user to run the DELETE USER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.DUPE</td>
<td>Allows a user to sign on at a second terminal but forces a signoff at the original terminal. Any transaction that is currently attached at the original terminal is also purged.</td>
</tr>
<tr>
<td>TOOLKIT.GPID</td>
<td>Allows a userid to be used as a group-id so that more than one user can use it to sign on to CICS.</td>
</tr>
<tr>
<td>TOOLKIT.LDSD</td>
<td>Allows a user to run the LISTDATASET command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.LGRP</td>
<td>Allows a user to run the LISTGROUP command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.LUSR</td>
<td>Allows a user to run the LISTUSER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.PEMT</td>
<td>Allows a user to run the PERMIT command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.RACL</td>
<td>Allows a user to run the RACLINK command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.RALT</td>
<td>Allows a user to run the RALTER command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.RDEF</td>
<td>Allows a user to run the RDEFINE command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.RDEL</td>
<td>Allows a user to run the RDELETE command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.REMV</td>
<td>Allows a user to run the REMOVE command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.RLIST</td>
<td>Allows a user to run the RLIST command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>Resource name</td>
<td>What it allows if a user has read access</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TOOLKIT.USRL</td>
<td>Allows a user to list usrdatal fields as part of the USRDAT command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.USRA</td>
<td>Allows a user to add and update usrdatal fields as part of the USRDAT command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.USRD</td>
<td>Allows a user to delete usrdatal fields as part of the USRDAT command of zSecure CICS Toolkit.</td>
</tr>
<tr>
<td>TOOLKIT.SPEC</td>
<td>Gives a user the equivalent of SPECIAL, when running RACF commands from zSecure CICS Toolkit. Users who have access to TOOLKIT.SPEC are allowed access to all resources within a specific command. For instance, if users have access to the LISTUSER command and TOOLKIT.SPEC, they can list any user and are not restricted by the LUSR.dftgrp definitions. While TOOLKIT.SPEC gives a user access to all resources within a command, it does not give the user access to the command. In this example, for the user to use the LISTUSER command, the user still requires access to TOOLKIT.USR. This applies only to RACF commands issued by the zSecure CICS Toolkit interface.</td>
</tr>
<tr>
<td>TOOLKIT.SVC</td>
<td>Allows a region to use the zSecure CICS Toolkit SVC.</td>
</tr>
<tr>
<td>ADGRgrpname</td>
<td>Defines the groups for which a user can issue an ADDGROUP.</td>
</tr>
<tr>
<td>ADUSdftgrp</td>
<td>Defines the default groups for which a user can issue an ADDUSER.</td>
</tr>
<tr>
<td>ALGRgrpname</td>
<td>Defines the default groups for which a user can issue an ALTGROUP.</td>
</tr>
<tr>
<td>AUSRdftgrp</td>
<td>Defines the groups for which a user can issue an ALTUSER.</td>
</tr>
<tr>
<td>CONNgrpname</td>
<td>Defines the groups for which a user can issue a CONNECT.</td>
</tr>
<tr>
<td>DELD.hlq</td>
<td>Defines the high-level-qualifier of data set names for which a user can issue a DELETE DATASET. See “Deleting a data set (DELETE DATASET command)” on page 50 for more details.</td>
</tr>
<tr>
<td>DELGgrpname</td>
<td>Defines the groups for which a user can issue a DELETE GROUP.</td>
</tr>
<tr>
<td>DELUdftgrp</td>
<td>Defines the default groups for which a user can issue a DELETE USER.</td>
</tr>
<tr>
<td>LGRPgrpname</td>
<td>Defines the groups for which a user can issue a LISTGROUP.</td>
</tr>
<tr>
<td>LUSRdftgrp</td>
<td>Defines the default groups for which a user can issue a LISTUSER.</td>
</tr>
<tr>
<td>PEMTdftgrp</td>
<td>Defines the default groups/groups for which a user can issue a PERMIT.</td>
</tr>
<tr>
<td>PEMXcdtclass</td>
<td>Defines the general resource classes, including the DATASET class, for which a user can issue a PERMIT, if the resource is not in a class that is defined in the CICS SIT.</td>
</tr>
<tr>
<td>PSWDDftgrp</td>
<td>Gives authority to change a user's PASSWORD within the specified default groups.</td>
</tr>
<tr>
<td>RACLdftgrp</td>
<td>Defines the groups for which a user can issue a RACLINK.</td>
</tr>
<tr>
<td>Resource name</td>
<td>What it allows if a user has read access</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RALT.cdtclass</td>
<td>Defines the general resource classes for which a user can issue an RALTER.</td>
</tr>
<tr>
<td>RDEF.cdtclass</td>
<td>Defines the general resource classes for which a user can issue an RDEFINE.</td>
</tr>
<tr>
<td>RDEL.cdtclass</td>
<td>Defines the general resource classes for which a user can issue an RDELETE.</td>
</tr>
<tr>
<td>REMV.grpname</td>
<td>Defines the groups for which a user can issue a REMOVE.</td>
</tr>
<tr>
<td>RLIST.cdtclass</td>
<td>Defines the general resource classes for which a user can issue an RLIST.</td>
</tr>
<tr>
<td>SECL.nnn</td>
<td>Specifies the SECLEVEL a user can specify for userid for which the user is authorized. The nnn is the SECLEVEL number from 001 to 254.</td>
</tr>
<tr>
<td>USRU.dftgrp</td>
<td>Defines the default groups for which a user can display USRDATA fields.</td>
</tr>
<tr>
<td>USRN.usrdnname</td>
<td>Defines the usrdnname that can be displayed or added/updated/deleted by the user.</td>
</tr>
</tbody>
</table>

**Automatic assignment of USS UIDs (OMVS AUTOUID)**

If you want to use the automatic assignment of unique USS UIDs, you must ensure that the following requirements are fulfilled.

- You must have the fix for APAR OW52135 installed.
- The RACF database must be enabled for Application Identity Mapping. The minimum stage that is required is stage 2.
- The profile BPX.NEXT.USER in the FACILITY class must be defined, with appropriate APPLDATA. The details are described in the RACF Security Administrator’s Guide. See the chapter RACF and z/OS UNIX.
- RACF TSO command usage of AUTOUID also requires that profile SHARED.IDS is defined and that the UNIXPRIV resource class is active and RACLISTEd.

**Automatic creation of home directories (OMVS MKDIR)**

If you want to use the automatic creation of the home directory when you create an OMVS segment for a user, the following extra RACF requirements must be fulfilled.

- The CICS region userid must have an OMVS segment that assigns a UID (or you must have enabled the use of a DEFAULT UID).
- The current group of the CICS region userid must have an OMVS segment that assigns a GID. Or, you must have enabled the use of a DEFAULT GID.
- The CICS region userid must have sufficient access to create the home directories. It can be implemented by one of the following methods:

| UID=0      | This option gives the CICS region full control over the entire USS environment. It is acceptable during initial testing, but is not suitable for a regular production environment. |
| UNIXPRIV   | You can also grant CONTROL access to the UNIXPRIV profile                                              |
SUPERUSER.FILESYS because it gives the CICS region READ/WRITE access to all files in the file system. Do not use this option for production usage.

**WRITE access to the directory where user home directories must be created.**
This option gives the CICS region exactly the authorization that is required. It is the preferred option.

When you select not to grant UID=0 to the CICS region userid, you must also give the CICS region the authority to set the correct owner for the newly created home directories. Failure to do so might render the new home directories unusable to the intended users. Set the correct owner with the CHOWN command, which typically requires authorization:

**UNIXPRIV SUPERUSER.FILESYS.CHOWN**
Granting READ access to this profile allows changing the owner (userid = dfltgrp) for all files in the system. Because it is a rather powerful authorization, it is better not to use this approach.

**UNIXPRIV CHOWN.UNRESTRICTED**
This discrete profile enables all users to change the owner of files/directories that they own. It is like the way RACF behaves for conventional data set profiles. Because zSecure CICS Toolkit creates the home directories initially with the CICS region as owner, it is authorized to change the owner to the intended user.

---

**zSecure CICS Toolkit restart**

Normally, after doing all the required updates and definitions for the installation, you must restart the CICS system to activate all the changes. For the initial installation, the restart is required to activate, for example, the SRT definitions. You can combine this activation with the system IPL that is required to pick up the definition of the zSecure CICS Toolkit SVC.

zSecure CICS Toolkit internally uses MVS subtasks. Normally, these subtasks are started during CICS initialization by a PLT program. These MVS subtasks are detached again at CICS termination by a second PLT program. Restarting CICS is the preferred method for this part of the installation process.

If all the required definitions are in place, but restarting CICS is not possible, you can use the following alternative method instead.

**Attention:** If you use this process to do the initial activation of zSecure CICS Toolkit, you must also use this process to deactivate it before you shut down the CICS system. Failure to do so results in A03 abends, probably followed by a system memory dump.

If the zSecure CICS Toolkit SVC is not installed, and the CICS SRT definition is not activated, attempts to install the software subtasks might result in a terminating abend of the CICS Started Task.

Sometimes, an error occurs during execution of the zSecure CICS Toolkit programs. Some of these errors result in the termination of one of the MVS subtasks that are used for zSecure CICS Toolkit processing. Current versions of the zSecure CICS Toolkit provide a transaction for stopping and restarting the MVS subtasks. Previous versions require a manual process. This manual process is described in [“Manual restart of the zSecure CICS Toolkit subtasks” on page 21](#).
**RTST transaction definition**

The RTST transaction can be defined to run the CRTKSTRT program. This program performs the necessary functions to stop and restart the zSecure CICS Toolkit subtasks. It also provides a function for refreshing the various zSecure CICS Toolkit programs, maps, and parameter modules.

When the RTST transaction is run, the following panel is displayed:

![RTST Transaction Panel]

---

**Figure 1. RTST Transaction panel**

This panel provides the following functions:

- **PF1** De-Activate subtasks
- **PF2** Activate subtasks
- **PF4** Refresh (newcopy) modules

---

Because the RTST transaction does not perform any internal authorization verification, access to the transaction must be controlled. Access must be restricted to those people who must be able to stop and start the zSecure CICS Toolkit subtasks, or refresh the modules.

**Manual restart of the zSecure CICS Toolkit subtasks**

When the RTST transaction is not available, you might also use the manual process to stop and start the zSecure CICS Toolkit subtasks. This process involves the use of the CECI transaction.

Run the following two commands:
When running these transactions, the terminal user must have access to CSML in DCICSDCT. If you chose to configure a different DESTID in CQTPCNTL, CSML must be replaced by your selected DESTID.

Use of the CICS Transaction Server with zSecure CICS Toolkit

There are several specifications to be aware of if you are running the CICS Transaction Server.

Be aware of the following specifications:

- The RSRCLASS parameter in CQTPCNTL must specify the correct RACF resource class. It does not default to the XTRAN parameter defined in the SIT.
- If you want to use the DUPEUSER capabilities of zSecure CICS Toolkit, it is best to start program CQTPSNP0. You can start this program in one of the following ways:
  - By doing an XCTL or LINK to the program
  - By starting the program as a transaction from your own signon program
  - From the signon exit point

The major purpose for using the DUPEUSER support is the ability to automatically cancel an existing session when you log on to a second terminal.

Note: Starting with CICS TS version 2.1, the behavior of the EXEC CICS SIGNON command changed. The new ID becomes effective only after termination of the current transaction. Module CQTPSNP0 in version 1.8.1 has more functionality to obtain the ID of the user that was signed on and verify the access to the TOOLKIT.DUPE and TOOLKIT.GPID resources. Other zSecure CICS Toolkit functions use the authorization of the user that started the signon transaction, and do not use the new ID.

Globalization

The BMS mapsets that are used by zSecure CICS Toolkit are provided as samples in SCQTSAMP.

An installation might modify these BMS mapsets, if the new mapsets are compatible with the existing mapsets. The copybook (symbolic map) resulting from the modification must be identical to the unmodified copybook that is used by the zSecure CICS Toolkit program modules. It can be done by retaining all fields and their lengths. The only parts that must be changed are field attributes (for example to suppress display of a certain field), or field initial values. Although CICS might allow other parts of the BMS maps to be changed, it is not supported in zSecure CICS Toolkit. Changed mapsets must be translated with the standard DFHMAPS procedure.
Chapter 3. Parameters for zSecure CICS Toolkit

The parameter module CQTPCNTL is used to define the parameters that zSecure CICS Toolkit uses.

After the parameters are set, use the RTCK transaction to check them. Errors are displayed with an error message. Resolve all errors before you implement zSecure CICS Toolkit.

Note: The contents of CQTPCNTL are revised and are not compatible with the contents of member CRTKCNTL used in the precursor product Consul zToolkit.

The following is an example of setting up the CQTPCNTL parameters:

```
CQTPCNTL CSECT
CQTPCNTL AMODE 31
CQTPCNTL RMODE ANY
*
EXITPGM DC CL8' ' 
DESTID DC CL4'CSML'
CICSAPPL DC CL8'IGNOREIT'
RSRCCLASS DC CL8'TCICSTRN'
CMNDPFX DC CL8'TOOLKIT.'
SMFUID DC CL8' '
SVCNUM DC CL3'222'
DUPEUSER DC CL1'2'
RACFOMND DC CL1'Y'
LOGGING DC CL1'Y'
PEMTALL DC CL1'Y'
LGDFLTU DC CL1'N'
END
```

Use the supplied CQTJCNTL sample job to apply the update from SMP/E. If that is not possible, you can also use the IBM supplied procedures DFHASMVS and DFHLNKVS to assemble and link the module. The parameters must be in the order as shown in the example. The definitions in the example are the defaults in the installation modules. You can customize the parameters for your installation.

Parameter Descriptions

The following list describes the parameters present in CQTPCNTL.

EXITPGM

The name of a program that is to receive control whenever the main zSecure CICS Toolkit transaction (normally RTMM) terminates. If you do not want to use an exit program, specify blanks. For a complete description of the exit program features, see Chapter 6, “zSecure CICS Toolkit exit points specifications,” on page 85.

DESTID

The destination zSecure CICS Toolkit uses for writing run time messages. The default is CSML. It can be changed to any other entry but must conform to the definitions for CSML as specified in the CICS Resource Definition manual.

CICSAPPL

This parameter is used as a prefix for the transaction names when you use the RSRC API interface for application security. If IGNOREIT is specified, or
the parameter is left blank, the parameter is ignored. If the parameter is coded, it can be up to eight bytes in length and must conform to RACF naming conventions. For more information, see Chapter 4, “Application security management,” on page 29.

RSRCCLASS
The RACF resource class that is used by zSecure CICS Toolkit. zSecure CICS Toolkit uses this class when it checks resources. It can be any of the classes that are defined in the SIT to CICS so that it is RACLSTed by CICS. It must be the name of the MEMBER class, not the group class (for example, TCICSTRN and not GCICSTRN). If you leave it blank or specify an invalid class, zSecure CICS Toolkit fails to initialize. The resource class to be used by the API when performing resource access checking (extended) can be defined as a parameter being passed to the API and overrides this definition for those resource checks. See the API documentation for further information.

CMNDPFX
The zSecure CICS Toolkit commands that a user can use, and that are displayed when the user enters the zSecure CICS Toolkit transaction (RTMM), are determined by the RACF definitions as outlined in step 1 in “Defining the RACF profiles” on page 14. These definitions all have a prefix of TOOLKIT. Using the CMNDPFX parameter, it is possible to specify a different prefix. However, this prefix is only in effect when the zSecure CICS Toolkit transaction is being used. When zSecure CICS Toolkit is being accessed through the API a prefix of TOOLKIT is always used.

For example, if CMNDPFX is CICSONE and a user wants to do an ADDUSER, the user must have:

- Access to CICSONE.A0US if they are using the zSecure CICS Toolkit transaction
- Access to TOOLKIT.A0US if they are doing the ADDUSER through the API

This provides more flexibility to do things with the API and restrict the options on the zSecure CICS Toolkit transaction display.

This parameter must be eight characters, must end with a period (.) and cannot contain any blanks. The default is TOOLKIT.

SMFUID
When zSecure CICS Toolkit creates an SMF record that reflects a change to the RACF database, the ID of the user logged on at the terminal is used in the SMF records. However, there might be situations where you want the ID in the SMF record to be different. For instance, if the SMF record is being shipped to another system to update another RACF database, the user might not have the required authority on that system to allow the change to occur. You can specify in SMFUID a different ID that is used in the SMF records created by zSecure CICS Toolkit.

If this parameter is left blank, the ID of the terminal user is used in the SMF records.

Note: This value is different from the SMF80UID field. The SMF80UID field is always set to the value TOOLKIT* to indicate that the record was produced as part of a zSecure CICS Toolkit function.

SVCNUM
The SVC number that was assigned to the zSecure CICS Toolkit SVC.
DUPEUSER
Use this parameter to control user signons.
- If this parameter is coded as 0, no checking takes place when a user signs-on.
- If it is a 1, zSecure CICS Toolkit checks to see whether the same RACF user ID is signed on at another terminal. If it is, the duplicate signon is disallowed and the user ID is signed off.
- Coding this parameter as 2 has the same effect as 1 with the exception that the terminal is logged-off as well (the equivalent of performing doing a CSSF LOGOFF).

Checking for duplicate user IDs is only done in the terminal that owns region when it is run in an MRO environment. If a user uses the CRTE transaction to route to an application region, no checking takes place in that region.

Permitting users:
When the DUPEUSER parameter is specified as 1 or 2, you have the option of allowing specific user IDs to be used as group IDs, or you can specify that certain IDs can log on at a second terminal but will be logged off at the terminal at which they are currently signed on. To use this capability, define the following resources to RACF. (This example assumes that the XTRAN parameter in the SIT is CICSTRN):
RDEFINE TCICSTRN (TOOLKIT.GPID TOOLKIT.DUPE) UACC(NONE)

To allow a user ID to be used as a group ID (that is, it can be shared by multiple users), permit it access to TOOLKIT.GPID.
For example:
PERMIT TOOLKIT.GPID CLASS(TCICSTRN) ID(GROUP01) ACCESS(READ)
This allows GROUP01 to be used by more than one user when they sign on to CICS.

To allow a user to log on at a second terminal, but to be forced off the first terminal, permit it access to TOOLKIT.DUPE.
For example:
PERMIT TOOLKIT.DUPE CLASS(TCICSTRN) ID(USER01) ACCESS(READ)
This enables USER01 to sign on at a second terminal but forces a signoff at the first terminal at which the user was signed on.

If a user has access to both TOOLKIT.DUPE and TOOLKIT.GPID, TOOLKIT.GPID takes precedence.

RACFCMND
If zSecure CICS Toolkit is only used by applications for internal security checking, controlling signons, or both, it does not require all of its subtasks in the region. These subtasks are only required if you use zSecure CICS Toolkit or the API to run RACF commands. It saves approximately 40K per region and results in an even greater savings if you have zSecure CICS Toolkit in multiple regions.
Specify Y if you use the commands or N if you do not.

LOGGING
When an application program uses the API to check resource access, zSecure CICS Toolkit produces SMF records based on the AUDIT
parameters for the resource. In other words, when a user does not have access to the resource and auditing for failures is turned on or AUDIT is all, zSecure CICS Toolkit produces SMF records.

This can result in a large amount of unwanted SMF records being produced. If you want zSecure CICS Toolkit to produce these records, specify Y. If you want them to be suppressed, specify N. For the RSRC and RSRX functions, it is also possible to specify S. This results in suppression of possible access violation messages, while still creating SMF records about these violations. For all other functions, S is interpreted the same as Y.

This parameter setting does not apply to SMF records produced by zSecure CICS Toolkit when a user alters or updates a profile on the RACF database.

PEMTAL
This parameter is used to control the scope of the PERMIT command. The PERMIT command can be restricted to giving access only to resources that are in a class that is specified in the SIT (for example; XDCT, XFCT, XJCT, SPPT, XTRAN), and is RACLISHED by CICS. In this case specify N. To allow the PERMIT command to be used for ALL resource classes, including DATASET, specify Y. For more information, see the PEMX and PEMT definitions in the "Internal Security Resource Listing" on page 16 and "Granting or removing access to a resource (PERMIT command)" on page 71.

LGDFLTU
This parameter is used to control the display of USERIDs as a subfunction of the LISTGROUP function. If this value is set to N, all users that are connected to the specified GROUP are shown. If this value is set to Y, only users that are connected to the DFLTGRP of the terminal user are shown.

CQTPCNTL parameter values verification
Each region can have its own version of CQTPCNTL. However, select the CICSAPPL value carefully to avoid errors.

After coding CQTPCNTL, use transaction RTCK to verify its parameters. Perform this action before zSecure CICS Toolkit is implemented.

Display the RTCK transaction panel. The following panel is displayed and shows any errors that might have occurred.
Select the function for the task you want to accomplish:

- Press PF01 from the main menu to view YOUR access to zSecure CICS Toolkit commands.
- Press PF02 from the main menu to view the zSecure CICS Toolkit programs, their status and PTF level.
- Press PF04 from the main menu to view the zSecure CICS Toolkit subtasks, their status and PTF level.
- Press CLEAR or PF03 to terminate the transaction. Any other key re-displays the main menu.

The following two panels show example output for the programs and subtasks.

**Sample partial panel showing the output for the program status:**

<table>
<thead>
<tr>
<th>Program</th>
<th>PTFlevl ST</th>
<th>Program</th>
<th>PTFlevl ST</th>
<th>Program</th>
<th>PTFlevl ST</th>
<th>Program</th>
<th>PTFlevl ST</th>
<th>Program</th>
<th>PTFlevl ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>CQTPOAPRM</td>
<td>OK</td>
<td>CQTPOAPRM</td>
<td>OK</td>
<td>CQTPOAPRM</td>
<td>OK</td>
<td>CQTPOAPRM</td>
<td>OK</td>
<td>CQTPOAPRM</td>
<td>OK</td>
</tr>
<tr>
<td>CQT PATCH</td>
<td>HCQT190</td>
<td>CQT PATCH</td>
<td>HCQT190</td>
<td>CQT PATCH</td>
<td>HCQT190</td>
<td>CQT PATCH</td>
<td>HCQT190</td>
<td>CQT PATCH</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00000</td>
<td>HCQT190</td>
<td>CQT00000</td>
<td>HCQT190</td>
<td>CQT00000</td>
<td>HCQT190</td>
<td>CQT00000</td>
<td>HCQT190</td>
<td>CQT00000</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00040</td>
<td>HCQT190</td>
<td>CQT00040</td>
<td>HCQT190</td>
<td>CQT00040</td>
<td>HCQT190</td>
<td>CQT00040</td>
<td>HCQT190</td>
<td>CQT00040</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00080</td>
<td>HCQT190</td>
<td>CQT00080</td>
<td>HCQT190</td>
<td>CQT00080</td>
<td>HCQT190</td>
<td>CQT00080</td>
<td>HCQT190</td>
<td>CQT00080</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00084</td>
<td>HCQT190</td>
<td>CQT00084</td>
<td>HCQT190</td>
<td>CQT00084</td>
<td>HCQT190</td>
<td>CQT00084</td>
<td>HCQT190</td>
<td>CQT00084</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00100</td>
<td>HCQT190</td>
<td>CQT00100</td>
<td>HCQT190</td>
<td>CQT00100</td>
<td>HCQT190</td>
<td>CQT00100</td>
<td>HCQT190</td>
<td>CQT00100</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00113</td>
<td>HCQT190</td>
<td>CQT00113</td>
<td>HCQT190</td>
<td>CQT00113</td>
<td>HCQT190</td>
<td>CQT00113</td>
<td>HCQT190</td>
<td>CQT00113</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00131</td>
<td>HCQT190</td>
<td>CQT00131</td>
<td>HCQT190</td>
<td>CQT00131</td>
<td>HCQT190</td>
<td>CQT00131</td>
<td>HCQT190</td>
<td>CQT00131</td>
<td>HCQT190</td>
</tr>
<tr>
<td>CQT00140</td>
<td>HCQT190</td>
<td>CQT00140</td>
<td>HCQT190</td>
<td>CQT00140</td>
<td>HCQT190</td>
<td>CQT00140</td>
<td>HCQT190</td>
<td>CQT00140</td>
<td>HCQT190</td>
</tr>
</tbody>
</table>

**Figure 3. zSecure CICS Toolkit: Program status output**

Sample partial panel showing the output for the subtask status:
<table>
<thead>
<tr>
<th>Program</th>
<th>PTFlev1 ST</th>
<th>Program</th>
<th>PTFlev1 ST</th>
<th>Program</th>
<th>PTFlev1 ST</th>
<th>Program</th>
<th>PTFlev1 ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>CQTS010</td>
<td>HCQT190 AV</td>
<td>CQTS020</td>
<td>HCQT190 AV</td>
<td>CQTS030</td>
<td>HCQT190 AV</td>
<td>CQTS040</td>
<td>HCQT190 AV</td>
</tr>
<tr>
<td>CQTS045</td>
<td>HCQT190 AV</td>
<td>CQTS050</td>
<td>HCQT190 AV</td>
<td>CQTS055</td>
<td>HCQT190 AV</td>
<td>CQTS060</td>
<td>HCQT190 AV</td>
</tr>
<tr>
<td>CQTS065</td>
<td>HCQT190 AV</td>
<td>CQTS070</td>
<td>HCQT190 AV</td>
<td>CQTS080</td>
<td>HCQT190 AV</td>
<td>CQTS090</td>
<td>HCQT190 AV</td>
</tr>
<tr>
<td>CQTS100</td>
<td>HCQT190 AV</td>
<td>CQTS110</td>
<td>HCQT190 AV</td>
<td>CQTS120</td>
<td>HCQT190 AV</td>
<td>CQTS130</td>
<td>HCQT190 AV</td>
</tr>
<tr>
<td>CQTS140</td>
<td>HCQT190 AV</td>
<td>CQTS150</td>
<td>HCQT190 AV</td>
<td>CQTS160</td>
<td>HCQT190 AV</td>
<td>CQTS170</td>
<td>HCQT190 AV</td>
</tr>
<tr>
<td>CQTS180</td>
<td>HCQT190 AV</td>
<td>CQTS190</td>
<td>HCQT190 AV</td>
<td>CQTS200</td>
<td>HCQT190 AV</td>
<td>CQTS210</td>
<td>HCQT190 AV</td>
</tr>
</tbody>
</table>

Figure 4. zSecure CICS Toolkit: Subtask status output
Chapter 4. Application security management

zSecure CICS Toolkit lets you request access to multiple resources with a single system call.

Traditionally, applications that run under CICS use some form of their own internal security. Even though CICS and external security manager might control access to the transactions, access to subfunctions of those transactions was inevitably maintained by the application.

This strategy causes the following undesirable results:

- Each application has a different form of security. Any user that requires access to multiple applications must frequently request access from multiple control points.
- Delays occur in granting access, and there is a lack of control as to who has what.
- Workloads increase.
- A single user often needs multiple IDs.

Since several years, applications also have the option to issue an EXEC CICS QUERY SECURITY function. However, if an application must establish authorization to many resources (for example, for determining which options to display on a selection panel), considerable time might be involved in requesting the authorization. In that situation, zSecure CICS Toolkit provides advantages. With the zSecure CICS Toolkit RSRC or RSRX functions, you can request access to multiple resources with a single system call.

Operator ID or OPID check

The traditional method that is employed by various applications to do security checking is based on the 3-byte operator ID of a user. This ID is checked against a table or a file, which contains an array or matrix of functions that this user, or operator ID, can perform.

Such internal security methods leads to numerous exposures. CICS provides no capability to ensure that operator IDs are unique. And the three characters that are used for the OPID are often not sufficient to accommodate all users. By using zSecure CICS Toolkit for application security checking, you can overcome these exposures. Other added advantages are the centralization of security definitions, and one security system (RACF) being used by all applications.

Application conversion

To use zSecure CICS Toolkit to check the OPID, you must convert existing applications.

This requires a coding change to the application that now checks security. If the application is a package, contact the vendor and create exit point within the package, where the security checking takes place. Use the zSecure CICS Toolkit API for security checksto use RACF security facilities. With the API, more than 2000 resources can be checked with one call. For more information about using the API and its multiple functions and capabilities, see Chapter 7, “Application Security”.
Application programs can link to CQTPAPI0, with a parameter that specifies the function that the user is attempting to access. CQTPAPI0 then checks to see whether the user has access to this function and returns to the calling application with the appropriate return code. This example shows how an application would use CQTPAPI0 and the definitions that would be made to RACF and the possible return codes from CQTPAPI0.

If a transaction that is run by a user had multiple functions, it would probably provide the user with a menu panel. The user would then select one of those options. In the following example, the user ran transaction ABCD. The application then gave the user an option menu as follows:

```
OPTION DESCRIPTION
1    READ PAYROLL MASTER RECORDS
2    UPDATE PAYROLL MASTER RECORDS
3    ADD PAYROLL MASTER RECORDS
4    DELETE PAYROLL MASTER RECORDS
```

Figure 5. Option menu example

The same transaction, ABCD, performs all the functions, but not all users must have access to all of them. To define the individual functions to RACF, an alias is assigned to each function. These functions can be READ for option one, UPDT for option two, ADDS for option three, and DELT for option four. After the user selected one of these options, the application program would LINK to CQTPAPI0 with a COMMAREA. This is described in the following example, which would contain one of the aliases (READ, UPDT, ADDS, or DELT). The return code would be set by CQTPAPI0.

```
* MVC API_FUNC,=CL4'RSRC' MOVE FUNCTION CODE FOR A RESOURCE CHECK
* MVC API_RESOURCE_NAME,=CL13'READ' OPTION REQUESTED BY USER
* MVI API_END,X'FF' END OF RESOURCE NAMES
* EXEC CICS LINK PROGRAM('CQTPAPI0') COMMAREA(API-COMM)
* CLI API_RESOURCE_RC,X'00' ACCESS ALLOWED ?
  BE ACCESSOK YES
  B  ERROR NO
*
* API_COMM   DS 0CL99
  API_FUNC   DS CL4  FUNCTION CODE
  API_RC     DS XL1  RETURN CODE
  API_MSG    DS CL79 MESSAGE AREA
  *
  API_RESOURCE_NAME DS CL13 RESOURCE NAME
  APIRESOURCE_RC  DS XL1 RACF RETURN CODE
  API_END       DS XL1 X'FF' END OF LIST
*
```

The return codes would be a one-byte hexadecimal code with the following meanings:

RETURN CODE MEANING:

X '00'  Access is allowed to resource
X '94'  The resource or classname is not defined to RACF
**Alias definitions**

The aliases, or mnemonics, that are used for application security checking, are defined to RACF in the same way as transactions.

When you define application aliases, they must be prefixed with the value defined in CQTPCNTL for CICSAPPL. If CICSAPPL is IGNOREIT or blank, a 13-byte alias would then be defined to RACF without a prefix. It must be padded to the right with blanks. The RACF definitions must be entered with the same RACF class name as the one used for zSecure CICS Toolkit. It is as specified in the RSRCLASS parameter of CQTPCNTL.

If the API_FUNCTION is RSRX, the resource name can be up to 246 characters in length. However, it must not be greater than the maximum defined for the resource class in the Class Descriptor Table.

Assuming the value that is coded for CICSAPPL was PRODAPPL, and the RSRCLASS value of TCICSTRN, the aliases would be defined as follows:

```
RDEFINE TCICSTRN (PRODAPPL.READ PRODAPPL.UPDT PRODAPPL.ADDS PRODAPPL.DELT)
```

Users would then be permitted to the resource as required:

```
PERMIT PRODAPPL.READ CLASS(TCICSTRN) ID(USER01) ACCESS (READ)
PERMIT PRODAPPL.UPDT CLASS(TCICSTRN) ID(USER02) ACCESS (READ)
PERMIT PRODAPPL.ADDS CLASS(TCICSTRN) ID(USER03) ACCESS (READ)
PERMIT PRODAPPL.DELT CLASS(TCICSTRN) ID(USER03) ACCESS (READ)
```

No definitions are required in any CICS tables. After the RACF profiles are created, zSecure CICS Toolkit is ready to check application security.

---

**Simple application security interface**

zSecure CICS Toolkit provides a direct interface as alternative to the full API provided by CQTAPI0 as described in Chapter 5.

The simple interface provides only two functions:

- Access the user profile, current group profile, and instdata of the currently signed on user
- Check access to a resource.

These two functions can also be accessed from modern CICS services like EXEC CICS ADDRESS ACEE and EXEC CICS QUERY SECURITY. To assist existing customers in their migration to these standard CICS services, the CQTPAPPL interface is provided.

**User information retrieval**

To use the user information retrieval function, you must provide a commarea of at least 22 bytes. If the commarea is larger than 286 bytes, the DFTLGRP and the INSTDATA for the user are returned as well.
To use this function, the first 4 bytes of the commarea must contain the value '????' (that is, four question marks). Here is an example of how to use this function:

```
MVI COMMAREA,"" Clear commarea
MVC COMMAREA+1(L'COMMAREA-1),COMMAREA
MVC COMM_FUNC,=CL4'????'
EXEC CICS LINK PROGRAM('CQTPAPPL')
COMMAREA(COMMAREA)
LENGTH(COMALEN)
CLI COMM_RC,X'00' Ok?
```

Further processing

```
COMAREA DS 0CL286 Space for COMMAREA
COMM_FUNC DS CL4 Function code ????
COMM_RSVD DS CL9 Unused
COMM_RC DS X11 Return code
COMM_USER DS CL8 Userid
COMM_GRP DS CL8 Group
COMM_IDL DS X11 Length of instdata
COMM_IDA DS CL255 Instdata
COMALEN DC AL2(*-COMMAREA) Length of COMMAREA
```

The return code has two possible values:
- X'00' user information is returned
- X'10' Invalid or no ACEE present.

If the commarea is too small, the value '*' (asterisk) is returned in the first byte of COMM_RESN. If the commarea has length zero, or is absent, no information is returned.

**Resource access verification**

To use the resource access verification function, you must provide a commarea of at least 14 bytes, containing the name of a resource.

The access of the current user is verified to be at least READ and the COMM_RC is set accordingly. The resource name must be left-aligned and padded with blanks. When processing the request, the name of the resource is prefixed with the value specified in CICSAPPL in CQTPCNTL. If the value of CICSAPPL is IGNOREIT or blank, no prefix is applied. The resource name as used in the access verification request is returned in the field COMM_RESN. Here is an example of how to use this function:

```
MVI COMMAREA,"" Clear commarea
MVC COMMAREA+1(L'COMMAREA-1),COMMAREA
MVC COMM_RESN,=CL13'PAYROLL'
EXEC CICS LINK PROGRAM('CQTPAPPL')
COMMAREA(COMMAREA)
LENGTH(COMALEN)
CLI COMM_RC,X'00' Access?
```

Further processing

```
COMAREA DS 0CL14 Space for COMMAREA
COMM_RESN DS CL13 Resource name
COMM_RC DS X11 Return code
COMALEN DC AL2(*-COMMAREA) Length of COMMAREA
```

The return code has three possible values:
- X'00' Access is allowed to resource
- X'04' The resource or classname is not defined to RACF
• X'08' The user is not authorized to use the resource.

If the commarea is too small, the value '*' (asterisk) is returned in the first byte of COMM_RESN. If the commarea has length zero, or is absent, no information is returned.
Chapter 5. The zSecure CICS Toolkit command interface

zSecure CICS Toolkit provides the capability of issuing RACF commands from CICS.

The commands that can be issued are:

ADDGROUP (ADGRP), ADDUSER (ADUSER), ALTGROUP (ALTGRP), ALTUSER (ALUSER), CONNECT (CONNC), DELETE GROUP (DELGRT), DELETE USER (DELUER), LISTDSD (LDS), LISTGRP (LGRT), LISTUSER (LUSER), PASSWORD, PERMIT, REMOVE, RALTER, RDEFINE, RDELETE, REMOVE, RLIST, RACLIN, USRDATA, and VERIFY.

For any option used to alter profiles, SMF records are produced, indicating the changes and who changed. These records show up in your normal RACF reports.

The SMF records all show the value TOOLKIT* in the SMF80UID field. This special value is used to indicate that the record was produced as part of a zSecure CICS Toolkit function.

The PASSWORD and VERIFY commands are only available through the API. You can use the zSecure CICS Toolkit API to customize the panels for your installation. For more information, see the API documentation.

Before users can use zSecure CICS Toolkit, they must be given access to one or more zSecure CICS Toolkit commands, as described in Chapter 2, “zSecure CICS Toolkit installation,” on page 3.

Navigating the Main menu

To view the command interface main menu, you must have access to the transaction to execute it. If you have not completed a signon to CICS with a valid RACF user ID, the transaction automatically terminates with message CQT006.

Procedure

1. To view the command interface main menu, enter RTMM at a clear panel.
2. To make a selection, press the PF key, or type in the number of the command you want to use and press Enter.

   When displaying a field within a profile (for example, the groups a user is connected to), you can use PF08 to page down if there is more than one panel of entries.

Adding, altering, or deleting a group (ADDDGROUP, ALTGROUP, or DELGROUP command)

Use the ADDGROUP, ALTGROUP, and DELGROUP commands to add a new group to the system, or to alter or delete an existing group.

About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.ADGR / TOOLKIT.ALGR / TOOLKIT.DELG / TOOLKIT.LGRP, depending on the command that is performed) and the group (ADGR.grpname / ALGR.grpname / DELG.grpname / LGRP.grpname).

Procedure

1. Access the ADDGROUP, ALTGROUP, and DELGROUP commands by pressing the designated PF key, as shown on the main menu.
2. Specify the values that are needed to perform the selected task. The different fields and their meanings are described under the LISTGROUP command. The fields that are required when you perform the different commands are as follows:

**ADDGROUP**
To define a new group, the GROUP name must be entered. This GROUP name must be unique and not currently exist as a group or user name. The OWNER, if not entered, defaults to your user ID. If a group name is entered as the owner, it must be the same name as the superior group (SUPGROUP). If you do not enter a SUPGROUP, it defaults to your current connect group. TERMUACC must be Y or N. UNIVERSAL must also be Y or N. The INSTALLATION DATA field is optional. After you entered the required information press PF01 to add the new group.

**ALTGROUP**
To alter the GROUP profile. You can change any of the fields except the GROUP. After you entered the data that you want to change, press PF04 to update the profile.

**DELGROUP**
To delete the GROUP profile. Enter the GROUP name that you want to delete, and press PF03. The group must not have any subgroups, users that are connected to it, or any group data sets. zSecure CICS Toolkit has no way to find all the users that might be connected to the group. Therefore, the group might not be a Universal group. zSecure CICS Toolkit checks for subgroups and users but not for group data sets.

---

**Adding a user profile (ADDUSER command)**

Use the **ADDUSER** command to add a user profile to the system.
About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.ADUS) and the default group of the user profile that is added (ADUS.dfltrgp).

Procedure

1. Access the ADDUSER command by pressing the designated PF key, as shown on the main menu.

![ADDUSER panel](image)

Figure 8. ADDUSER panel

2. Enter information in these fields to add a user:

   USERID
   In the ADDUSER= field. USERID can be 1 - 8 characters.

   NAME
   Is normally the users name, 1- 20 characters.

   DFLTGRP
   The user default group. It must be a valid group name.

   AUTHORITY
   The authority of the user within the group. The default is U (use), but might be changed to C (create).

   SECLEVEL
   Specify the SECLEVEL for the user or leave blank. Available SECLEVELs can be displayed with the RLIST command and displaying class SECDATA, resource name SECLEVEL and then selecting the MEMBERS display.

   LOGON DAYS
   Enter a Y or N to indicate which days the user can access the system.
   Specifying N for any particular day prevents the user from accessing the system on that day.

   LOGON TIME
   LOGON TIME specifies the time of day the user might log on. Leave
the FROM and TILL fields zero to allow the user to log on at any time. If a time is specified, it must be in the range 0001 through 2359.

**PASSWORD**
The initial password for the user. The initial password is always set as expired.

**OWNER**
The owner of the profile.

**PASSWORD PHRASE**
The initial password phrase for the user. The initial password phrase is always set as expired. Trailing blanks are removed.

**Note:** Password phrases are only supported on z/OS 1.8 and higher. Attempts to set a password phrase on z/OS levels that do not support them result in message CQT184. The user is defined as specified but without a password phrase.

**INSTALLATION DATA**
Information about the user can be entered in this field. If data is entered, the EOF key must be pressed after the last character was entered. This field is optional.

3. Press PF05 to add the user to the system.

The initial PASSWORD for the user, if not specified, can be the same as the name of the DEFAULT GROUP. Users must enter a new password the first time they log on.

### Changing a profile (ALTUSER command)

Use the **ALTUSER** command to change the profile for a specific user.

**About this task**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the users default group (AUSR.dfltgrp). SMF records are created any time that a profile is altered by any user.

**Procedure**

1. Access the **ALTUSER** command by pressing the designated PF key, as shown on the main menu.
2. Specify the user for whom you want to change the profile.
   When you display a user, you are given the option to resume the user or not.

3. To reset a user password to the name of the DEFAULT GROUP, clear the PASSWORD field: place the cursor in the first position of the field, press EOF, then press PF05.

4. To set a specific PASSWORD, clear the field, place the cursor in the first position of the field, enter the new password and press PF05. By default the new password is set to expired. If you have the special attribute, you can indicate that the new password does not need to be changed immediately (set Expire PW to N).

   Starting with z/OS 1.8, users might also be issued a password phrase. If you must change a password phrase, specify the new value. Ensure that you remove excess characters at the end. zSecure CICS Toolkit strips trailing blanks, but trailing question marks (?) or asterisks (*) are included as part of the password phrase. If you want to remove an existing password phrase, blank out or erase the entire password phrase field. If you want to retain the current value, leave all the question marks as shown.

5. You can REVOKE a user one of two ways by specifying REVOKEDT:
   - Set REVOKEDT to the current date, the equivalent of specifying ALTUSER userid REVOKE. The revoke flag is set and the REVOKE and RESUME dates are cleared.
   - Specify a REVOKEDT date other than current date. The user is revoked on that date.

6. Use RESUMEDT to resume a user. You can use the special value 00000 (five zeros) to remove any REVOKE or RESUME date. Setting the field to blanks or empty, or leaving the value as shown, results in retaining the current value. Specify Y for Resume User to resume a user immediately.

7. To update one of the supported segments, press the indicated PF key.
   - To modify CICS information for the user, press PF06.
   - To update the TSO segment, press PF07.
   - For the OMVS and WORKATTR segments, use PF08 and PF09.
8. Unless you have SPECIAL, or access to TOOLKIT.SPEC, you are not permitted to change any field below the delimiter line of ++++++. There are exceptions for the DFLTGRP value the RESTRICTED attribute. Regular administrators can change these two fields without the need for SPECIAL. Another exception is the EXPIRED setting for a new password. The user needs system special or access to TOOLKIT.SPEC to set non-expired passwords. If you do have SPECIAL, you can alter and of the fields on the panel and press PF05 to update.

9. Specify an entry in the CLAUTH or NOCLAUTH to give or remove class authority in the specified class.

zSecure CICS Toolkit does not verify that the specified class is valid.

**Altering the CICS segment for a user (ALTUSER CICS SEGMENT)**

Use the **ALTUSER** command with the CICS SEGMENT option to alter the CICS segment for a specific user.

**About this task**

The user must have the following authorizations:

- The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the user default group (AUSR.dfltgrp).
- To manage the CICS segment, the user must have access to TOOLKIT.ACIC. Version 1.4 (the precursor Consul zToolkit product) enforced this requirement only if the TOOLKIT.ACIC profile was defined or covered by a generic profile.
- Version 1.8.1 and higher of Tivoli zSecure CICS Toolkit or zSecure CICS Toolkit require access to resource TOOLKIT.ACIC.

**Procedure**

1. To access the ALTUSER CICS SEGMENT command, press the PF06 key in the main ALTUSER panel.

![Figure 10. ALTUSER (CICS SEGMENT) panel](image)
OPIDENT
The one-to-three character operator identification to assign to this user.

OPPRTY
The operator priority for this user. It can be in the range 000 - 255.

TIMEOUT
The number of minutes that must elapse since the user last used the terminal before CICS timeouts the terminal.

For releases of RACF before 2.2, it must be a value in the range 000 - 255. For later releases of RACF, the range is 000 - 999. In both cases, a value of zero means the terminal is not timed out.

XRFSOFF
The CICS extended recovery facility sign-off option. Specify FORCE to sign off the operator in the event of XRF takeover, or NOFORCE to leave the operator signed on.

OPCLASS
Operator classes are used by CICS when it routes basic mapping support messages. The valid classes are in the range 01 - 24. When you specify the operator classes, they must be separated by a comma (for example: 01,04,05,16,24).

RSLKEY
The RSL keys are used by CICS on distributed platforms. Each CICS resource has one RSL key that is assigned to it. In order for a user to access a resource, the user must have the same RSL key as the RSL key assigned to the resource. The valid keys are in the range 01 - 24. The values 00 and 99 have special meaning. When you specify RSLKEYs, they must be separated by a comma, for example, 01,04,05,16,24.

Note: In the current release, zSecure CICS Toolkit provides space for only 22 RSLKEY values.

TSLKEY
The TSL keys are used by CICS on distributed platforms. Each CICS transaction has one TSL key that is assigned to it. In order for a user to run a transaction, the user must have the same TSL key as the TSL key assigned to the transaction. The valid keys are in the range 01 - 64. The values 00 and 99 have special meaning. When you specify TSLKEYs, they must be separated by a comma, for example, 01,04,05,16,24.

Note: In the current release, zSecure CICS Toolkit provides space for only 22 TSLKEY values.

2. To display the current information in the CICS SEGMENT of a user, enter the userid, and press Enter.

3. To change any or all of the information, enter the new data and press PF5. If there are any errors, an error message displays, indicating the problem.

4. To remove the CICS SEGMENT, press PF11.

Note: Removing a CICS SEGMENT does not prevent a user from accessing CICS services. Access to CICS services might be controlled by profiles in the APPL resource class.

Without a CICS segment, CICS users inherit certain values from the CICS SEGMENT of the CICS Default User.
Whenever the OPCLASS, RSLKEY or TSLKEY parameter is updated, it completely replaces the prior values. For example, if a user has OPCLASS 01,02,03 defined, and you update the profile by specifying 02,05,06, the user only has 02, 05, and 06 defined. The prior values of 01,02, and 03 are deleted.

**Altering the TSO segment for a user (ALTUSER TSO SEGMENT)**

Use the ALTUSER command with the TSO SEGMENT option to alter the TSO segment for a specific user.

**About this task**

The user needs the following authorizations:

- The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the user default group (AUSR.dfltgrp).
- To manage the TSO segment, the user must have access to TOOLKIT.ATSO. Version 1.4 (the precursor Consul zToolkit product) enforced this requirement only if the TOOLKIT.ATSO profile was defined or covered by a generic profile.
- Version 1.8.1 and higher of Tivoli zSecure CICS Toolkit or zSecure CICS Toolkit require access to the TOOLKIT.ATSO resource.

**FUNCTION**

**AUTHORITY:**

**Procedure**

1. To access the ALTUSER TSO SEGMENT command, press the PF07 key in the main ALTUSER panel.

```
Termid = CP24    IBM Security zSecure CICS Toolkit    Date = 2007/094
Userid = BCSCGB1 ALTUSER = BCSCGB2    Time = 11:11:05
Acctnum = * Destid =
HClass = JClass = MsgClass = SClass =
Size = 0000000 Maxsize = 0000000 Seclabl =
Proc = ISPFPROC Unit = Udata = 0000
```

Figure 11. ALTUSER (TSO SEGMENT) panel

**ACCTNUM**

The users default TSO account number.

**DESTID**

The default destination for dynamically allocated SYSOUT data sets.
HCLASS
The default hold class of the user.

JCLASS
The default job class of the user.

MSGCLASS
The default message class of the user.

SCLASS
The default sysout class of the user.

SIZE
The minimum region size if the user does not request one at logon time.

MAXSIZE
The maximum region size the user can request at logon time.

SECLABL
The users security label.

PROC
The name of the user default logon procedure.

UNIT
Default name of a device or group of devices that a procedure uses for allocations.

UDATA
Installation data for the user.

2. To display the current information in the TSO SEGMENT of a user, enter the userid, and press Enter.

3. To change any or all of the information, enter the new data and press PF5.
   If there are any errors, an error message displays, indicating the problem. If a field is set to blanks, that parameter is deleted from the user TSO segment.
   When the parameters ACCTNUM, PROC and SECLABL are specified, the user must have access to these definitions in the appropriate RACF resource class.
   Refer to the RACF Command Language Reference for complete information and the other TSO segment fields.

4. To remove the TSO SEGMENT, press PF11.

Note: Removing a TSO SEGMENT prevents the user from accessing TSO interactive services.

Altering the OMVS segment for a user (ALTUSER OMVS SEGMENT)

Use the ALTUSER command with the OMVS SEGMENT option to alter the OMVS segment for a specific user.

About this task

The user needs the following authorizations:
- The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the user default group (AUSR.dfltgrp).
- For managing the OMVS segment, the user must have access to TOOLKIT.AOMV.
**Procedure**

1. To access the **ALTUSER OMVS SEGMENT** command, press the **PF08** key in the main ALTUSER panel.

   ![Figure 12. ALTUSER OMVS segment panel](image)

**UID**

The OMVS UID of the user. When you change the UID to another value, it is possible to enter AUTOUID in this field. If the required profiles are defined, zSecure CICS Toolkit assigns the next available UID. When you change the UID into a value that is already assigned to another user, the command is rejected. For authorized users, it can be overridden by usage of the SHARED parameter.

**Shared**

When you assign a UID to a user, the UID must be unique. When the terminal user has the System-SPECIAL attribute or has access to SHARED.IDS in the UNIXPRIV class, the user might request that the UID value can be shared between multiple users. In other words, the UID value then does not need to be unique. If you want to assign a shared UID, enter **Y** in the SHARED field.

**MKDIR**

When you assign a HOME directory to a user, the directory must exist in the file system. By selecting option **Y**, the zSecure CICS Toolkit task attempts to create the directory, and set the owner to the userid and dfltgrp. The authority to run the necessary USS commands is based on the CICS region user, and not on the authority of the CICS terminal user. If this function is not enabled in your installation, leave the value for this field as **N**.

**Home**

The home directory of the user. When you change the OMVS segment of the user, ensure that the case of this field is correct. Either do not update the OMVS segment at all, or ensure that your terminal uses mixed case. Or, verify that the actual home directory of the user is defined in ALL UPPERCASE. If the HOME directory (case sensitive) cannot be located, use of UNIX System Services might fail.
Program
The initial program (shell program) for the user. When you change the
OMVS segment of the user, ensure that the case of this field is correct.
Either do not update the OMVS segment at all, or ensure that your
terminal uses mixed case. Or, verify that the program actually exists in
ALL UPPERCASE. If you leave the field blank, USS normally uses the
value /bin/sh as default value.

ASSizeMax
The address-space-size that you define is a numeric value 10 485 760 - 2
147 483 647. The value that is specified overrides any value that is
provided by the MAXASSIZE parameter of BPXPRMxx. If the system
value is adequate, you must leave this field blank.

CPUMax
The processor time that you define is a numeric value 7 - 2 147 483 647.
The value that is specified overrides any value that is provided by the
MAXCPUMAX parameter of BPXPRMxx. If the system value is
adequate, you must leave this field blank.

FileProcMax
The files-per-process that you define is a numeric value 3 - 524287.
Regular users can use the value 256. The value that is specified
overrides any value that is provided by the MAXFILEPROC parameter
of BPXPRMxx. If the system value is adequate, you must leave this
field blank.

MMapAreaMax
The memory-map-size that you define is a numeric value 1 - 16 777
216. The value that is specified overrides any value that is provided by the
MAXMMAPAREA parameter of BPXPRMxx. If the system value is
adequate, you must leave this field blank.

ProcUserMax
The processes-per-UID that you define is a numeric value 3 - 32 767.
The value that is specified overrides any value that is provided by the
MAXPROCUSER parameter of BPXPRMxx. If the system value is
adequate, you must leave this field blank.

ThreadsMax
The threads-per-process that you define is a numeric value 0 - 100 000.
Specifying a value of 0 prevents applications that are run by this user
from using the pthread_create service. The value that is specified
overrides any value that is provided by the MAXTHREAd parameter
of BPXPRMxx. If the system value is adequate, you must leave this
field blank.

MemLimit
The nonshared-memory-size that you define to RACF is a numeric
value 0 - 16777215, followed by the letter M, G, T, or P.

SHMemMax
The shared-memory-size that you define to RACF is a numeric value 1
- 16777215, followed by the letter M, G, T, or P. The value that is
specified for SHMEMMAX overrides any value that is provided by the
IPCSHMEMSEG parameter of BPXPRMxx. If the system value is
adequate, you must leave this field blank.

2. To display the current information in the OMVS SEGMENT of a user, enter the
userid, and press Enter.
3. To change any or all of the information, enter the new data and press PF5. If there are any errors, an error message displays, indicating the problem. You can delete a field from the user's OMVS segment by setting the value to all blanks. For setting the UID to the value zero (0), the terminal user must have the RACF System-SPECIAL attribute. Access to TOOLKIT.SPEC is not applicable to this particular function. Use the value AUTUID whenever possible. The AUTUID function is only available in z/OS 1.4 and higher. It requires definition of the BPX.NEXT.USER profile in the facility class. For more information, see Chapter 2, “zSecure CICS Toolkit installation,” on page 5.

4. To remove the OMVS SEGMENT, press PF11.

Note: Removing an OMVS SEGMENT prevents the user from accessing any UNIX System Services. Access to USS might also be provided by the default UID designated by BPX.DEFAULT.USER in the FACILITY class.

Altering the WORKATTR segment for a user (ALTUSER WORKATTR SEGMENT)

Use the ALTUSER command with the WORKATTR SEGMENT option to alter the WORKATTR segment for a specific user.

About this task

The user needs the following authorizations:

- The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the user default group (AUSR.dflgrp).
- For managing the WORKATTR segment, the user must have access to TOOLKIT.AWRK.

Procedure

1. To access the ALTUSER WORKATTR SEGMENT command, press the PF09 key in the main ALTUSER panel.
Name Specifies the name of the user SYSOUT information is to be delivered to.

Account Specifies an account number for APPC/MVS processing. Although RACF accepts any string of up to 255 characters, the zSecure CICS Toolkit interface allows up to 60 characters.

Bldg Specifies the building that SYSOUT information is to be delivered to.

Dept Specifies the department that SYSOUT information is to be delivered to.

Room Specifies the room SYSOUT information is to be delivered to.

Addr1 Address-line-1 specifies other address line for SYSOUT delivery.

Addr2 Address-line-2 specifies other address line for SYSOUT delivery.

Addr3 Address-line-3 specifies other address line for SYSOUT delivery.

Addr4 Address-line-4 specifies other address line for SYSOUT delivery.

2. To display the information in WORKATTR SEGMENT of a user, enter the userid, and press Enter.

3. To change any or all of the information, enter the new data and press PF5. If there are any errors, an error message displays, indicating the problem. If a field is set to blanks, that parameter is deleted from the users WORKATTR segment.

When you change the WORKATTR segment of the user, ensure that the case of these fields is correct. If your installation requires mixed case values in these fields, you might refrain from updating the WORKATTR segment at all, or ensure that your terminal uses mixed case.

4. To remove the WORKATTR SEGMENT, press PF11.

Note: Removing a WORKATTR SEGMENT normally does not affect the users of their authorization to use any system services.
Connecting a user or group to a group (CONNECT command)

Use the CONNECT command to connect a user or group to a group.

About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.CONN) and to the target group (CONN.grpname).

Procedure

1. To access the CONNECT command, press the designated PF key on the main menu.
2. To connect a user to a group, enter the user and group name as indicated and press PF05. If there are any errors, such as an invalid user or group name, an error message indicates the problem.

[Table]

<table>
<thead>
<tr>
<th>Termid = CP24</th>
<th>IBM Security zSecure CICS Toolkit</th>
<th>Date = 2007/094</th>
</tr>
</thead>
<tbody>
<tr>
<td>Userid = BCSCGB1</td>
<td>CONNECT</td>
<td>Time = 11:14:03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Connect</th>
<th>Group</th>
<th>Authority</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td></td>
<td>U</td>
<td>BCSCGB1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Special</th>
<th>Operations</th>
<th>Revokedt</th>
<th>Resumedt</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>N</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CQT016 - Enter userid and group name
PF5 = Update ENTER = Redisplay CLEAR = Main Menu

Figure 14. CONNECT panel

AUTHORITY
  Defaults to U (use) but might be set to C (create), N (connect), or J (join).

SPECIAL
  Specify Y if the user must have the group-special attribute.

OPERATIONS
  Specify Y if the user must have the group-operations attribute.

OWNER
  Defaults to the ID of the person who is issuing the command, but any valid user ID or group can be entered.

REVOKEDT
  The date (YYDDD) the user is prevented from connecting to the group. If today’s date is specified, the connection is revoked immediately. In this case, the value of RESUMEDT is ignored, and both the RESUMEDT and the REVOKEDT are reset. The special value 00000
(five zeros) can be used to remove an existing REVOKEEDT. Setting the field to blanks or empty, results in leaving the current value unchanged.

**RESUMEDT**

The date (YYDDD) the user can connect to the group. If today's date is specified, the connection is resumed immediately. In this case, the value of REVOKEEDT is ignored, and both the RESUMEDT and the REVOKEEDT are reset. The special value 00000 (five zeros) can be used to remove an existing RESUMEDT. Setting the field to blanks or empty, results in leaving the current value unchanged.

# Deleting a data set (DELETE DATASET command)

Use the **DELETE DATASET** command to delete a data set profile from the system.

**About this task**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.DELD) and the high-level-qualifier of the data set profile name (DELD.hlq). If the user does not have access to the DELD.hlq, standard RACF authority checking is used. Refer to the RACF Command Language Reference manual for information about which data set profiles a user is authorized to delete.

**Procedure**

To access the **DELETE DATASET** command, press the designated PF key on the main menu.

---

**Termid** = CP24  IBM Security zSecure CICS Toolkit  **Date** = 2007/094  **Userid** = BCSCGB1  **Time** = 11:14:18  **Dsname**  **Delete** =  **Generic** = Y

---

*CQT144 -Enter dataset profile to be deleted. Specify Y if Generic, N if not*

*PFS=Update ENTER=Redisplay CLEAR=Main menu*

---

*Figure 15. DELETE DATASET panel*

# Deleting a user profile

Use the **DELETE** command to delete a user profile from the system.
About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.DELU) and the default group of the user (DELU.dfltgrp).

Before the user profile is deleted, it must be REMOVED from any groups it is connected to except its default group. No data set profiles that have this userid as a high-level qualifier can exist.

zSecure CICS Toolkit checks for group connections but not for data set profiles.

Procedure

To access the DELETE command, press the designated PF key on the main menu.

Listing the profile for one or more data sets (LISTDSET command)

Use the LISTDSET command to list the profile for a specific data set or multiple data sets.

About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.LDSD).

Procedure

1. To access the LISTDSET command, press the designated PF key on the main menu.
LISTDSET

The ID of the data set to be displayed (if you perform a listuser). If a search is being done, this field can be bypassed, or any characters can be entered in any position as part of the search criteria.

**Note:** If you are doing a normal listdset, the following fields are not used for entry. These other fields are only used as entries if you are doing a search (PF11).

G,D, or * Allows you to search for generic (G), discrete (D), or both (*) types of profiles.

OWNER

A user or group that was defined as the owner of the data set.

CRE

The date the data set was created. The format is YYDDD.

LAST REF

The date the data set was last referenced. The format is YYDDD.

LAST CHG

The date the data set was last updated. The format is YYDDD.

UACC

The universal access for the data set. This field can be ALTER, CONTROL, UPDATE, READ, or NONE.

ALTER ACC

The number of times the data set was accessed with ALTER.

CNTRL ACC

The number of times the data set was accessed with CONTROL.

UPDTE ACC

The number of times the data set was accessed with UPDATE.

READ ACC

The number of times the data set was accessed with READ.

GROUP DS

This field can be Y or N.
WARN
Indicates whether the data set is in warning mode. This field can be Y or N.

CRE GROUP
The current connect group of the user that created this data set.

DATASET TYPE
This field identifies the data set type. These first two characters in this field indicate a VSAM (VS), or Non-VSAM (NV) data set profile. The third character indicates whether the profile is a model profile (M) or not (N). Finally, the fourth character indicates whether the profile is for a tape data set (T) or not (N).

LEVEL
The level indicator for the data set. This field is a numeric field.

AUDIT
Indicates the audit flag for the data set. The settings can be: A to audit all accesses, S to audit successful accesses, F to audit failures, or N for no auditing.

AUD SUCC
The audit SUCCESS flag. The settings can be: R to audit successful reads, U to audit successful updates, C to audit successful control accesses, or A to audit successful alter accesses.

AUD FAIL
The audit FAILURE flag. The settings can be: R to audit unsuccessful reads, U to audit unsuccessful updates, C to audit unsuccessful control accesses, or A to audit unsuccessful alter accesses.

GLBL AUDIT
The Global audit options as specified by a user with the AUDITOR attribute. The settings can be: A to audit all accesses, S to audit successful accesses, F to audit failures, or N for no auditing.

GAUD SUCC
The GLOBAL audit SUCCESS flag. The setting can be: R to audit successful reads, U to audit successful updates, C to audit successful control accesses, or A to audit successful alter accesses.

GAUD FAIL
The GLOBAL audit FAILURE flag. The settings can be: R to audit unsuccessful reads, U to audit unsuccessful updates, C to audit unsuccessful control accesses, or A to audit unsuccessful alter accesses.

SECL
The security level of the data set. This field is a numeric field.

NUMCTGY
The number of security categories to which the data set belongs.

Numpgms
The number of programs that are authorized to access the data set.

Numusrs
The number of users and groups authorized to access the data set.

INSTALLATION DATA
The information that is contained in the data sets DATA field. This installation data can be up to 255 characters.

2. Press PF05 to display the access list of this data set profile (the users and groups).
3. Press PF07 to display the programs in the conditional access list.
4. Press PF01 to toggle the display (if you are doing a search) and display all the data sets that match the criteria.
5. Press PF03 to clear the fields and enter new criteria for a search or LISTDSET.

**LISTDSET Display Example**

You can view the programs in the conditional access list.

```
Termid = CP24  IBM Security zSecure CICS Toolkit  Date = 2007/094
Userid = BCSCGB1  Listdset =  Time = 11:15:07
G , D OR * G  SYS1.**
Owner = SYS1  Cre = 05033 Last ref = 05033 Last chg = 05033 Uacc = READ
Alter acc = 000000 Cntvl acc = 000000 Updte acc = 000000 Read acc = 000000
Group ds = Y WARN = N Cre grp = SYS1  Dataset type = NVNN Level = 000
Audit = F Aud Succ = R Aud Fail = R Gbl audit = N Gaud Succ = R Gaud Fail = R
Secl = *** Numctgy = 0000 NumPgms = 0000 NumUsrs = 0003

----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----

|<== |

----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----

PF1=Toggle 3=Chgopts 5=Users 7=Programs 11=Search CLEAR=Main menu
```

*Figure 18. LISTDSET Display panel*

Now you can choose to do one of the following:
- Display access list entries (PF05, "userids").
- Display conditional access list entries (PF07, "programs").
- Change the search or list options (PF03).
- Return to the main menu (CLEAR).
- If you are doing a search, display all data sets that meet the criteria (PF01).

**Toggling the LISTDSET panel**

If you are doing a search, you can toggle the panel and display all the data sets that match the criteria.

**Procedure**

- To do a search, press PF01. All data sets that match the criteria are displayed.
Choose a PF key for the task you want to do:
- Change the search or list options (PF03).
- Return to the main menu (CLEAR).
- If the panel is full with data sets, display the next panel (ENTER).
- To perform a LISTDSET on any ID, enter the ID in the LISTDSET field and press PF01.

**Viewing the users authorized, their access authority and access count (LISTDSET USERIDS)**

Use the LISTDSET user IDs option to display the users authorized to access the data set, the access authority of the user, and their access count.

**Procedure**
- Press PF05 from a LISTDSET panel.
Choose a PF key for the task you want to do:
- Change the search or list options (PF03).
- Display the programs that can access this data set (PF07).
- Return to the LISTDSET panel (PF09).
- Display the next data set if your are doing a search (ENTER).
- Return to the main menu (CLEAR).

**Viewing the program/userid combination (LISTDSET Programs)**

Use the LISTDSET panel to display the program/userid combination that is authorized to access the data set and to display the access authority.

**Procedure**
- Press PF07 from a LISTDSET panel.
Choose a PF key for the task you want to do:
- Change the search or list options (PF03).
- Display the access list entries (PF05).
- Return to the LISTDSET panel (PF09).
- Display the next data set if you are doing a search (Enter).
- Return to the main menu (CLEAR).

Listing the profile for one or more groups (LISTGROUP command)

Use the LISTGROUP command to list the profile for a specific group or multiple groups.

About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.LGRP) and the target group (LGRP.grpname).

Procedure
1. To access the LISTGROUP command, press the designated PF key on the main menu.
LISTGROUP
The ID of the group to be displayed (if performing a listgroup). If a search is being performed, this field can be bypassed, or any characters can be entered in any position as part of the search criteria.

Note: If you are performing a normal listgroup, the fields are not used for entries. These other fields are only used as entries if you are doing a search (PF11).

SUPGROUP
The superior group to this group.

OWNER
A user or group that has been defined as the owner of this group.

UNIV
An indicator if it is a Universal group. The list of users connected to a Universal group only shows those users that have non-standard authorizations within the group.

CRE
The date this profile was created. The format is YYDDD.

UACC
The authority of a user to the group if the user is not connected to the group. This field can be JOIN, CONNECT, CREATE, USE or NONE. This field cannot be set using any RACF command or zSecure CICS Toolkit. It must have a value NONE for all groups, except for the fixed group VSAMDSET.

TERMUACC
Indicates if a group or user must be explicitly authorized to access a terminal. This field can be Y or N.

NUMBER OF SUBGROUPS
The number of subgroups to this group. This field is a numeric field.

NUMBER OF USERS
The number of users connected to this group. This field is a numeric field. For a Universal group, it only reflects the number of users that have non-standard authorizations within the group.
**MODEL**

The name of a discrete data set profile to be used as a model for a new *groupname* data sets. This field is an alphanumeric field.

**INSTALLATION DATA**

The information contained in the data sets DATA field. This information can be up to 255 characters.

2. You can either enter a specific group to perform a listgroup on that ID, or enter any character in the field as the search criteria.
   - After entering the search criteria, press **PF11** to start the search.
   - A normal listgroup is performed by entering the group name and pressing **Enter** to open the LISTGROUP panel.

3. Press **PF04** to display the users connected to this group with a DELETEUSER option.

4. Press **PF05** to display the users connected to this group.

5. Press **PF07** to display the subgroups.

6. Use **PF01** to toggle the display (if you are performing a search) and display all the groups that match the criteria.

7. Press **PF03** to clear the fields and enter new criteria for a search or LISTGROUP.

**LISTGROUP Display Example**

Enter a group to be listed or initiate a search to display the panel.

<table>
<thead>
<tr>
<th>Termid = CP24</th>
<th>IBM Security zSecure CICS Toolkit</th>
<th>Date = 2007/094</th>
</tr>
</thead>
<tbody>
<tr>
<td>Userid = BCSCGB1</td>
<td>LISTGROUP = SYSPROG</td>
<td>Time = 11:24:02</td>
</tr>
<tr>
<td>Supgroup = SYSL</td>
<td>Owner = SYSL</td>
<td>Univ = N Cre = 05033 Uacc = NONE</td>
</tr>
<tr>
<td>Termuacc = Y Number of subgroups = 00000 Number of users = 00003</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 23. LISTGROUP Display panel**

Now you can choose to:
- Display users (PF05).
- Display only users that are also connected to your default group.
- Display subgroups (PF07).
- Change the search or list options (PF03).
- Return to the main menu (CLEAR).
- If you are performing a search, display all groups that meet the criteria (PF01).
Toggling the LISTGROUP panel

If you are doing a search, you can toggle the LISTGROUP panel and displays all groups that match the criteria.

Procedure

• To perform a search, press PF01. All groups that match the criteria are displayed.

Listing users for a group (LISTGROUP command, USERIDS option)

You can use the LISTGROUP command with the USERIDS option to list all the users connected to a group.

Procedure

• Press PF05 from a LISTGROUP panel to display all the users connected to the group.
Choose a PF key for the task you want to perform:
- Change the search or list options (PF03).
- Display the alternate users display (PF04).
- Display the subgroups (PF07).
- Return to the LISTGROUP panel (PF09).
- Display the next group if performing a search (ENTER).
- Return to the main menu (CLEAR).

Deleting user IDs from a LISTGROUP

You can use delete one or more user IDs from the list of user IDs connected to a group.

Procedure
- Press PF04 from a LISTGROUP panel to display all the users connected to the group and the option to delete one or more of them.
Choose a PF key for the task you want to perform:

- Change the search or list options (PF03).
- Display the users display (PF05).
- Display the subgroups (PF07).
- Return to the LISTGROUP panel (PF09).
- Display the next group if performing a search (ENTER).
- Return to the main menu (CLEAR).

To delete selected users, enter a D next to the user profile you want to delete and then press PF11. Authority to delete the user profile is controlled by the standard zSecure CICS Toolkit DELUSER authorization.

**Listing the subgroups of a group**

You can list all the subgroups of a group.

**Procedure**

- Press PF07 from a LISTGROUP panel to display all the subgroups of the group.
Choose a PF key for the task you want to perform:

- Change the search or list options (PF03).
- Display the alternate users display (PF04).
- Display the users (PF05).
- Display the users also connected to your default group (PF06).
- Return to the LISTGROUP panel (PF09).
- Display the next group if performing a search (ENTER).
- Return to the main menu (CLEAR).

Listing the profiles for a user ID (LISTUSER command)

You can use the LISTUSER command to list the profile for a specific user ID.

About this task

FUNCTION

AUTHORITY

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.LUSR) and the users default group (LUSR.dftgrp).

Procedure

1. To access the LISTUSER command, press the designated PF key on the main menu.
LISTUSER

The ID of the user to be displayed (if performing a listuser). If a search is being performed, this field can be bypassed, or any characters can be entered in any position as part of the search criteria.

**Note:** If you are performing a normal listuser, the fields are not used for entries. These other fields are only used as entries if you are doing a search (PF11).

**NAME**

The user name. A maximum of 20 alphanumeric characters.

If you want to search for a name, or part of a name, anywhere within the name field, use the following format:

To search for SMITH anywhere in the name field enter <SMITH. It returns all profiles that have the characters SMITH anywhere in the name field. The <> indicates to zSecure CICS Toolkit that the criterion is a different search criterion for this field than if using the wild-card characters of "*".

**OWNER**

A user or group that has been defined as the owner of the user.

**PASSWORD**

Not used.

**CRE**

The date this profile was created. The format is YYDDD.

**DFLTGRP**

The name of the default group for the user.

**AUTHORITY**

This is the users authority within the default group. The possible entries for this field are **ASORGGAT**. The meanings of the subfields are: A indicates ADSP, GA indicates GROUP AUDITOR, and T indicates that terminal access is required.
**UACC**  The universal access of the user for the default group. This field can be
*ALTER, CONTROL, UPDATE, READ* or *NONE*.

**CLASSCNT**
The number of classes in which the user is allowed to define profiles.

**SPECIAL**
Indicates if the user has the SPECIAL attribute. This field can be *Y* or *N*.

**OPERATIONS**
Indicates if the user has the OPERATIONS attribute. This field can be *Y* or *N*.

**AUDITOR**
Indicates if the user has the AUDITOR attribute. This field can be *Y* or *N*.

**RESTR**
This field indicates if the UACC, GAC and ID(*) do apply for this user. This field can be *Y* or *N*.

**GRPACC**
Specifies that group data sets created by this user are accessible to other
users in the group. This field can be *Y* or *N*.

**ADSP**
Indicates that new data sets created by this user are automatically
protected by discrete profiles. This field can be *c*.

**PROTECTED**
This field indicates if the userid can be used by specification of the
password. PROTECTED userids can only be propagated, started, or
used through surrogate. This field can be *Y* or *N*.

**UAUDIT**
Indicates if all RACHECKs and RACDEFs issued for the user can be
logged. This field can be *Y* or *N*.

**REVOKE**
Indicates if the REVOKE attribute is set for the user. This field can be *Y*
or *N*.

**REVOKEDT**
The date that the user is revoked. The format is *YYDDD*.

**RESUMEDT**
The date the user is resumed. The format is *YYDDD*.

**LASTACC**
The date and time the user last accessed the system by using RACINIT.
The format is *YYDDD/HH:MM:SS*. If the user has never logged on, this
field contains ? in the first position.

**PASSDATE**
The date the users password was last changed. The format is *YYDDD*
or the field is zero if it has been reset.

**PASSINT**
The interval that the users password is in effect. This field is a numeric
field.

**PWTRY**
The number of unsuccessful password attempts by this user. This field
is a numeric field.
SECL  The security level of the user. This field is a numeric field.

SMTWTFSS  The days of the week that the user can logon. A 'Y' indicates the user can logon for that day, 'N' indicates the user is restricted for that day.

FROM  If the user is restricted by time, FROM is the starting time that the user might log on at. The format is HHMM. If there are no time restrictions, both FROM and TILL are 0000.

TILL  The latest time the user can logon to the system. The format is HHMM.

PWDGEN  The current password generation number for the user. This field is a numeric field.

PWDCNT  The number of old passwords present for this user. This field is a numeric field.

NUMCTGY  The number of security categories the user has access to. This field is a numeric field.

MODEL  The data set profile model for this user. This field is an alphanumeric field.

INSTALLATION DATA  The information contained in the users DATA field. This information can be up to 255 characters.

2. You can either enter a specific userid to perform a LISTUSER on that user, or enter any character in the field as the search criteria.
   • After entering the search criteria, press PF11 to start the search.
   • A normal LISTUSER is performed by entering the userid and pressing Enter.

3. Press PF05 to display the categories for this user.

4. Press PF07 to display the groups.

5. Use PF01 to toggle the display (if you are performing a search) and display all the users that match the criteria.

6. Press PF03 to clear the fields and enter new criteria for a search or LISTUSER.

**LISTUSER Display Example**
   This example shows what might be displayed if you specify a user ID and press Enter, or enter different criteria and press PF11 to initiate a search.

Note: If you enter a full user ID and click PF11, you obtain the same results as if you clicked Enter because there is only one profile with the specified user ID.
From this point, you can choose to:
- Display the categories (PF05).
- Display the groups (PF11).
- Change the search or list option (PF03).
- Return to the main menu (CLEAR).
- If you are performing a search, display all users that meet the criteria (PF01).

### Toggling the LISTUSER panel

You can use the LISTUSER panel to list all the users that match the criteria you enter.

**Procedure**
- To perform a search, press PF01. All users that match the criteria are displayed.

---

**Termid = CP24**

**Userid = BCSCGB1**

**LISTUSER = BCSCGB2**

**Name = Guus 2nd**

**Owner = BCSC**

**Password = ????????**

**Cre = 05033**

**Dfltgrp = BCSC**

**Authority =**

**Uacc = NONE**

**Classcnt = 0001**

**Special = N**

**Operations = Y**

**Auditor = N**

**Restr = N**

**Grpacc = N**

**Adsp = N**

**Protected = N**

**Uaudit = N**

**Revoke = N**

**Revokedt = *******

**Resumedt = *******

**Lastacc = 07089/09:17:57**

**Passdate = 07068**

**Passint = 180**

**PwTry = 00**

**Sec1 = *****

**SMWTS From Till**

**Pwdgen = 006**

**Pwdcnt = 006**

**NumCtgy = 0000**

**NumGrp = 0005**

**YYYYYY 0000 0000 Model =**

---

|<==|==|==|==|==|==|==|==|==|
|==|==|==|==|==|==|==|==|==|
|==|==|==|==|==|==|==|==|==|

**PF1=Toggle 3=Chgopts 5=Ctgy 6=Segments 7=Groups 11=Search CLEAR=Main menu**

*Figure 29. LISTUSER Display panel*
Choose a PF key for the task you want to perform:
– Change the search or list options (PF03).
– Return to the main menu (CLEAR).
– If the panel is full with userids, display the next panel (ENTER).

To perform a LISTUSER on any ID, enter the ID in the LISTUSER field and press PF01.

Listing groups for a user ID (LISTUSER command, GROUPS option)

You can use the LISTUSER command with the Groups option to display the groups that a user ID is connected to.

**Procedure**

• Press PF07 from a LISTUSER panel to display the groups that a user is connected to.
Choose a PF key for the task you want to perform:
- Change the search or list options (PF03).
- Display the categories for this user (PF05).
- Return to the LISTUSER panel (PF09).
- Display the next user if performing a search.
- Return to the main menu (CLEAR).

**Listing categories for a user ID (LISTUSER command, Categories option)**

You can use the LISTUSER command with the Categories option to list the categories that a user ID is connected to.

**Procedure**

- Press PF05 from a LISTUSER panel to display the categories that a user is connected to.
Choose a PF key for the task you want to perform:

- Change the search or list options (PF03).
- Display the groups for this user (PF07).
- Return to the LISTUSER panel (PF09).
- Display the next user if performing a search.
- Return to the main menu (CLEAR).

**Listing the TSO and CICS segments for a user ID (LISTUSER command, Segments option)**

You can use the LISTUSER command with the Segments option to list the TSO and CICS segments for a user ID.

**Procedure**

- Press **PF06** from a LISTUSER panel to display the TSO and CICS segments for the user.
Press PF08 to display the OMVS and WORKATTR segments. Pressing PF08 again scrolls up to the previous segment display.

**Figure 33. LISTUSER (Segments)**

- Choose a PF key for the task you want to perform:
  - Change the search or list options (PF03).
  - Display the groups for this user (PF07).
  - Return to the LISTUSER panel (PF09).
  - Display the next user if performing a search.
  - Return to the main menu (CLEAR).

**Figure 34. LISTUSER: OMVS and WORKATTR panel**

**Granting or removing access to a resource (PERMIT command)**

You can use the `PERMIT` command to grant or remove access to a resource.
About this task

The resource might be in:
1. One of the resource classes defined in the SIT for this run of CICS or
2. Any other general resource class or the DATASET class.

The user must have access to the zSecure CICS Toolkit command
(TOOLKIT.PEMT) and the default group of the user ID or group (PEMT.dflgrp)

and

If the resource is in a class defined in the SIT, the user must also have access to the
resource, at a level equal to or higher than the level of access that is being given.
After the PERMIT has been completed, the resource classes must be recreated in
order to have immediate effect. zSecure CICS Toolkit does not provide a way for
issuing the required SETROPTS REFRESH command.

If the resource is in any other class, the user must also have authority to issue
PERMIT commands in that class (PEMX.classname), and also to the resource, at a
level equal to or higher than the level of access that is being given.

Procedure

1. To access the PERMIT command, press the designated PF key on the main menu.

| Termid = CP24 | IBM Security zSecure CICS Toolkit | Date = 2007/094 |
| Userid = BCSCGB1 | Permit | Time = 11:35:10 |

- User/Grp =
- Rsrclass =
- Resource =

<<<<

- Delete = N
- Access = R (R=Read, N=None, U=Update, A=Alter, C=Control)
- Specify "Delete = Y" to remove the user from the access list

CQT029 - Enter userid/group name and resource
PF5=Update ENTER=Redisplay CLEAR=Main menu

Figure 35. PERMIT panel

USERID
The name of the user or group to be granted access (or removed).

RESOURCE
The name of the resource (for example, CEMT if it was the transaction CEMT).

RSRCLASS
The resource class name. If blank, the value of the XTRAN parameter
specified in the SIT is used.
DELETE
Specify Y in this field to remove a user or group from the access list for this resource.

ACCESS
Specify R for READ access or N for NONE, U for update, A for alter or C for control. If N is specified, the user must have a minimum of READ authority to issue the command.

2. Update or specify the information and press PF5. The user executing the PERMIT command must have access to the resource that is being altered. For example, if access is being given to CEMT, the user must have access to CEMT. If DELETE is specified as Y, the user executing the command must still have access to the resource. The level of access required is whatever specified in the ACCESS field.

Maintaining associations (RACLINK command)

You can use the RACLINK command to define, list, undefine, or approve user associations.

About this task

The RACLINK command only works with local nodes. Independent of the value of the NODE specified on the panel, zSecure CICS Toolkit assumes it to be the name of the local node.

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.RACL). To issue a RACLINK for a different user, the user must have RACF Special, TOOLKIT.SPEC or access to the default group of the user (RACL.dfltgrp).

Procedure

1. To access the RACLINK command, press the designated PF key on the main menu.
TYPE
The type of association. It can be PEER or MANAGED.

NODE
The name of the node where the association is defined.

USERID
The user that the association is being defined for.

PWD SYNC
Specifies if the association has password synchronization or not (enter YES or NO)

PASSWORD
Optional parameter. The password for the user specified.

2. In the field RACLINK =, enter the user for which the command is being issued. Press Enter to list the associations for that user.

3. Enter a U to the left of an association and then press PF05 to cause it to be undefined.
   - The association needs to be for your own ID, or you need system special, TOOLKIT.SPEC or RACL.dfltgrp.
   - If the association is awaiting approval and it is your own ID (or you have system special or zSecure CICS Toolkit special), you can approve it by entering A to the left of it and pressing PF05.
   - In the preceding display, a new association is about to be defined. After the user presses the PF5 key, the new association is defined. The D must be present to indicate a DEFINE.
   - The association is implicitly approved if a valid PASSWORD was provided or the issuer of the command has RACF SPECIAL, TOOLKIT.SPEC or RACL.dfltgrp.
   - To define a new user association, you need access to the normal RACLINK authorizations as defined for the TSO RACLINK command. These authorizations are RACLINK.DEFINE.nodename and RACLINK.PWSYNC.nodename.
Listing and maintaining profiles in a general resource class (RALTER / RDEFINE / RDELETE commands)

You can use the RALTER, RDEFINE, and RDELETE commands to list and maintain profiles in a general resource class defined in the CDT.

About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.RALT / TOOLKIT.RDEF / TOOLKIT.RDEL, depending on the command being performed) and to the general resource class (RLST.cdtclass in addition to RALT.cdtclass / RDEFCdtclass / RDEL.cdtclass).

Procedure

1. To access the RALTER / RDEFINE / RDELETE command, press the designated PF key on the main menu.

   **Termid = CP24**  IBM Security zSecure CICS Toolkit  **Date = 2007/094**
   **Userid = BCSCGB1**  **RES Class =**  **Type =**  **Time = 11:37:16**

   **Profile**

   **Member**

   **Owner =**  **Notify =**  **Uacc = None**  **Warn = N Level = 000**
   **Audit = F Aud succ = R Aud fail = R**
   
   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----
   |<==
   
   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----
   
   PF1=Rdef  PF2=Addmem  PF3=Rdel  PF4=Delmem  PF5=Updprof  ENTER=Rlst  CLEAR=Main menu

   **Figure 37. RALTER / RDEFINE / RDELETE panel**

**RDEFINE**

To define a new profile/resource, all fields are required, except for the installation data. After entering these information, press **PF01** to perform the **RDEFINE**.

**RDELETE**

To delete a profile/resource, enter the CLASS and PROFILE names and press **PF03**.

**RALTER**

Allows you to add a new member to a group (ADDMEM), delete a member from a group (DELMEM), or update the profile information (UPDPROF). The information required depends on the subcommand being performed.

**ADDMEM**

Requires the CLASS, PROFILE and MEMBER to be added. Press **PF02** to complete the ADDMEM.
DELMEM
Requires the CLASS, PROFILE and MEMBER to be deleted.
Press PF04 to delete the member.

UPDPROF
Requires all fields, except for the installation data. By entering
the CLASS and PROFILE fields and pressing Enter, all the
current entries for each field display. These entries can then be
over stepped. Press PF05 to complete the update.

2. Specify the values for the action you want to perform using the field
descriptions in step 1, then press the corresponding PF key to initiate the
change.

Removing user IDs or groups from a group (REMOVE command)

You can use the REMOVE command to remove user IDs or groups from a group.
User IDs cannot be removed from their default group.

About this task

The user must have access to the zSecure CICS Toolkit command
(TOOLKIT.REMV) and the target group (REMV.grpname).

Procedure

1. To access the REMOVE command, press the designated PF key on the main
 menu.

2. To remove a user from a group, enter the user and group name as indicated
 and press PF05. Users cannot be removed from their default group.

Listing the profiles for a general resource class (RLIST command)

You can use the RLIST command to list the profiles for a general resource class
defined in the CDT.
About this task

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.RLST) and the general resource class (RLST.cd1c1ass).

Procedure

1. To access the RLIST command, press the designated PF key on the main menu.

```
Termid = CP24  IBM Security zSecure CICS Toolkit  Date = 2007/094
Userid = BCSCGB1  Rlist class = Type = Time = 11:37:41

Owner = ********  Dte = *****  Last ref = *****  Last chg = *****  Uacc = *******
Audit = * Aud succ = * Aud fail = * Notify = ********  Warn = * Level = ***
Glb Audit = * Gaud Succ = * Gaud Fail = * Secl = ***
Members = ****  NumUsrs = ****  Condacc = ****

-+-1---2---3---Installation data-5---6---7---
-+-1---2---3---4---5---6---7---
-+-1---2---3---4---5---6---7---
PF1=Toggle 3=Chgopts 5=Members 7=Users 9=Condacc 11=Search CLEAR=Main menu
```

Figure 39. RLIST panel

RLIST CLASS

This is the name of the resource class to be displayed. It must be a valid entry in the RACF Class Descriptor Table.

TYPE

This defines the class as being a member (TYPE=M) or group (TYPE=G) class. It is provided as information only and is not used as input for a list or search.

PROFILE

The name of the profile to be displayed. Enter the name of the profile to be displayed in this field. If a search is being performed, this field can be bypassed, or any characters can be entered in any position as part of the search criteria. This field can be up to 246 characters in length.

Note: If you are only performing an RLIST, the rest of the fields are ignored. The remaining fields are only used as input when doing a search (PF11).

OWNER

A user or group that has been defined as the owner of the profile.

DTE

The date this profile was created. The format is YYDDD.

LAST REF

The date the data set was last referenced. The format is YYDDD.

LAST CHG

The date the profile was last updated. The format is YYDDD.
UACC  The universal access for the profile. This field can be ALTER, CONTROL, UPDATE, READ or NONE.

AUDIT  Indicates the audit flag for the profile. The settings can be: A to audit all accesses, S to audit successful accesses, F to audit failures, or N for no auditing.

AUD SUCC  This is the audit SUCCESS flag. The settings can be: R to audit successful reads, U to audit successful updates, C to audit control accesses, or A to audit successful alter accesses.

AUD FAIL  This is the audit FAILURE flag. The settings can be: R to audit unsuccessful reads, U to audit unsuccessful updates, C to audit unsuccessful control accesses, or A to audit unsuccessful alter accesses.

NOTIFY  The user to be notified when access is denied to this profile.

WARN  Indicates if the profile is in warning mode. This field can be Y or N.

LEVEL  The level indicator for the data set. This field is a numeric field.

GLBL AUDIT  The Global audit options as specified as by a user with the AUDITOR attribute. The settings can be: A to audit all accessed, S to audit successful accesses, F to audit failures, or N for no auditing.

GAUD SUCC  This is the GLOBAL audit SUCCESS flag. The settings can be: R to audit successful reads, U to audit successful updates, C to audit successful control accesses, or A to audit successful alter accesses.

GAUD FAIL  This is the GLOBAL audit FAILURE flag. The settings can be: R to audit unsuccessful reads, U to audit unsuccessful updates, C to audit unsuccessful control accesses, or A to audit unsuccessful alter accesses.

SECL  The security level of the profile. This field is a numeric field.

MEMBERS  The number of members in this profile, if it is a group profile.

NUMUSRS  The number of users and groups authorized to access the profile. This field is a numeric field.

CONDACC  The number of user/groups on the conditional access list. This field is a numeric field.

INSTALLATION DATA  The information contained in the DATA field of the profile. It can be up to 255 characters.

2. You can choose to:
   • Display the users/groups with access to the profile (PF07).
   • Display users/groups on the conditional access list (PF09).
   • If the resource class is a group class, as indicated by the TYPE= field, display the members in the profile (PF05).
When displaying members, users, or the conditional access list, you can use PF08 page down if there is more than one panel to be displayed.

**RLIST Display Example**

This example shows what is displayed when you enter a profile to be listed.

If you enter a profile to be listed, or initiate a search, the next panel is displayed as follows.

![Figure 40. RLIST Display panel](image)

From this point, you can choose to:
- Display the members (PF05).
- Display the users (PF07).
- Display the conditional access list (PF09).
- Change the search or list options (PF03).
- Return to the main menu (CLEAR).
- If you are performing a search, display all profiles that meet the criteria (PF01).

**Listing the members in a profile (RLIST command, MEMBERS option)**

You can use the **RLIST** command with the Members option to list the members in a profile.

**Procedure**
- Press PF05 from an RLIST panel to display the members in profile.
From this point, you can choose to:
- Display the users (PF07).
- Display the conditional access list (PF09).
- Change the search or list options (PF03).
- Return to the main menu (CLEAR).
- If you are performing a search, display the next profile (ENTER).

Listing user IDs in a profile and the access they have (RLIST command, USERS option)

You can use the RLIST command with the Users option to list the user IDs in the profile and the access they have.

Procedure
- Press PF07 from an RLIST panel to display the users in the profile and the access they have.
From this point, you can choose to:

- Display the members (PF05).
- Display the conditional access list (PF09).
- Change the search or list options (PF03).
- Return to the main menu (CLEAR).
- If you are performing a search, display the next profile (ENTER).

**Listing users/groups in the conditional access list for a profile (RLIST command, CONDACC option)**

You can use the **RLIST** command with the Conditional Access option to list the users/groups in the conditional access list for the profile.

**Procedure**

- Press **PF09** from an RLIST panel to display the user/groups in the conditional access list for the profile.
From this point, you can choose to:
- Display the members (PF05).
- Display the users (PF07).
- Change the search or list options (PF03).
- Return to the main menu (CLEAR).
- If you are performing a search, display the next profile (Enter).

Listing, adding, updating, or removing the USRDATE fields from a profile (USRDATA command)

You can use the **USRDATA** command to list, add, update, or remove the USRDATA fields from a user profile.

**About this task**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.USRL), the userid (USRU.dfltgrp), and the USRDATA name (USRN.usrdname). For the ADD, UPDATE and DELETE subfunctions, access to the corresponding command profile is required (TOOLKIT.USRA for ADD and UPDATE or TOOLKIT.USRD for DELETE).

**Procedure**

1. To access the **USRDATA** command, press the designated **PF** key on the main menu.
2. To display USRDATA of a user, enter the user ID and press *Enter*.
   - The USRDATA names and the first 64 characters of the corresponding values display on the bottom part of the panel.
   - If more USRDATA names are present than fit on the panel, press **PF8** to scroll down.
   - For an untruncated display of one USRDATA value, use the **S** (or **L**) command in front of the wanted USRDATA name/value and press **Enter**.

3. To add USRDATA for a user, enter the name and value of the USRDATA, then enter an **A** in the field in front of the **CLASS** and press **PF05**. You can also use
This same method to delete or update USRDATA fields from selection of a D or U in this command field. This latter method is an alternative to the preferred methods described here.

4. To delete one of the displayed USRDATA name/value pairs, use either of the following methods:
   - Use the D command in front of the USRDATA you want, and press PF05.
   - Use the S or L line command to display the USRDATA value, followed by PF11 on the detail panel.

5. To update existing USRDATA values, go to the detail panel obtained by using S (or L) in front of the listed USRDATA name. On the detail panel, type over the value with the new value and press PF05.
Chapter 6. zSecure CICS Toolkit exit points specifications

In the EXITPGM parameter in CQTPCNTL, you can specify a program that is to receive control whenever the main zSecure CICS Toolkit transaction (normally RTMM) terminates and returns control to CICS.

Control is passed to the EXITPGM through the XCTL command. zSecure CICS Toolkit does not receive control again after the EXITPGM, which includes a return code and the data, if any, that was sent to the user's panel. The format of the COMMAREA is as follows:

<table>
<thead>
<tr>
<th>EXITRC</th>
<th>DC CL1</th>
<th>Return Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>CICS Toolkit transaction has terminated</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Signon transaction failed or was terminated with the clear key</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Signon completed. User was signed-on at a second terminal but was not authorized (did not have access to TOOLKIT.DUPE and DUPEUSER checking is in effect).</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Signon completed. Same as 3, but terminal logged off CICS.</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Signon completed. No CICS segment found for user.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Signon completed. CICS segment was found for user.</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Signon completed. Error in installation data being used for operator information.</td>
<td></td>
</tr>
</tbody>
</table>

EXITDATA DC CL335 Data sent to users screen from signon or CICS Toolkit transaction termination. If the return code is 0, this field is only 79 bytes in length. For any other return code, this field will contain the data, if any, that was sent to the terminal user.

A sample EXITPG is provided in the SCQTSAMP pds as member CQTXSNEX.
Chapter 7. Application programming interface (API)

The zSecure CICS Toolkit Applications Programming Interface allows users to access the RACF database directly from a CICS application program. No special knowledge of RACF or its database format is required and the applications do not need to run authorized.

Using the API, an installation can tailor the zSecure CICS Toolkit panels to suit their own requirements, or produce different types or reports. zSecure CICS Toolkit ensures that only authorized users access the RACF database. The same rules apply when using the API as for using zSecure CICS Toolkit. The user executing a transaction that is using the API must be authorized to the zSecure CICS Toolkit commands used. If requesting user information, the user must have access to the default group of the user profile being displayed. For more information about this subject, see Chapter 5, “The zSecure CICS Toolkit command interface,” on page 35.

Using the API is a procedure that only requires the CICS application program to call the interface module, CQTPAPI0, and pass certain parameters to it in a COMMAREA. These parameters inform zSecure CICS Toolkit of the command being requested and also provide the storage area where the requested information is returned.

For compatibility reasons, the CQTPAPI0 program has an alias CRTKAPI. Although both names refer to the same module, use the new name, CQTPAIP0, in all applications.

For any option used to alter profiles, SMF records are produced, indicating the changes and who changed them. These records show up in your normal RACF reports.

The SMF records all show the value TOOLKIT* in the SMF80UID field. This special value is used to indicate that the record was produced as part of a zSecure CICS Toolkit function.

zSecure CICS Toolkit does not cause the CICS main task to wait while it is reading or updating the RACF database. All these commands are processed by the zSecure CICS Toolkit subtasks, leaving CICS free to continue with normal transaction processing. This CICS region does not run in an authorized state at any time, in keeping with the IBM statement of integrity.

Command requests using the COMMAREA

The COMMAREA is the way zSecure CICS Toolkit is informed of the command being requested. The size of the COMMAREA varies depending upon the command being executed. In all cases, there is a common header, followed by specific information for the relevant command.

The format of the header is as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API_FUNCT</td>
<td>CL4</td>
<td>This field specifies the command being requested.</td>
</tr>
<tr>
<td>API_RC</td>
<td>XL1</td>
<td>A one byte hexadecimal return code.</td>
</tr>
</tbody>
</table>
The message that would normally be displayed on the terminal if the user was using the standard Toolkit transaction is returned in this field.

The application program invokes the API using a standard CICS LINK command:

```cics
EXEC CICS LINK PROGRAM('CQTPAPI0') COMMAREA(APICOMM) LENGTH(APILEN)
```

If a COMMAREA is not passed, CTKAPI just returns to the caller without any further processing. If an error is detected in the COMMAREA layout, either the length or the command requested is invalid. Or, the user is not authorized to the zSecure CICS Toolkit COMMAND and the API sets API_RC with a return code, indicating the nature of the error. The possible return codes are:

**RETURN CODE**

X'00' COMMAREA is correct

X'01' An invalid command was requested. Check the parameter specified in API_FUNC and verify that it is correct.

X'02' The length of the COMMAREA is too small for the requested command.

X'03' The user is not authorized to use this zSecure CICS Toolkit COMMAND or is not signed on at the terminal.

X'04' There was no profile protecting the TOOLKIT, function resource. An authorization decision could not be made, and the API function was not executed.

X'05' Internal error, contact Technical Support

If the commarea passed to the API is correct, but the function fails for some other reason, the API_RC contains the value X'00'. The function-specific return code field (typically called API_function_RC) contains an error indicator. Most API-functions use the value -1 as indicator that an error occurred. The field API_MSG contains the error message describing the failure. Here are the examples of these messages:

- CQT039 is for the ALTUSER command if the specified ID does not exist.
- CQT080 is for the LIST commands if the requested profile cannot be found.

For the complete text of the error messages, see the IBM Security zSecure: Messages Guide.

All fields in the COMMAREA must be padded with blanks unless indicated otherwise in the documentation.

### Change the authorized user

To execute a command through the API, the userid associated with the task must be authorized to the zSecure CICS Toolkit command. This userid is normally the one for the user logged on at the terminal or the CICS default user. You might specify a different userid to be the authorized user.

To change the authorized user, make the following definition in the first 24 bytes of the API-MSG area before calling CQTPAPI0:
Table 7. API-MSG definition to change the authorized userid

<table>
<thead>
<tr>
<th>API Message Variable</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API_MSG_USERAUTH</td>
<td>DC CL8'USERAUTH'</td>
<td>A constant of USERAUTH</td>
</tr>
<tr>
<td>API_MSG_USERID</td>
<td>DC CL8'USERAUTH'</td>
<td>The USERID to be used as the authorized user.</td>
</tr>
<tr>
<td>API_MSG_PASSWORD</td>
<td>DC CL8'password'</td>
<td>The PASSWORD for the user.</td>
</tr>
</tbody>
</table>

If the password is incorrect for the user, the command fails and the appropriate message is returned to the calling program.

Perform a search

A search of the RACF database can be performed using the LISTUSER, LISTGROUP, LISTDATASET, and RLST commands.

The COMMAREA for each of these commands contains a one-byte code field. It indicates whether a search is being performed, if the next profile must be retrieved, or if other information about the profile is required.

When a search is requested, the profile attribute fields in the COMMAREA must be padded with asterisks. The search mask can be any combination of valid characters or letters in any combination of fields. When zSecure CICS Toolkit finds a match, it returns the profile information into the COMMAREA. To retrieve the next profile that matches the search criteria, set the code field in the COMMAREA to N (next) and call the API. The profile itself must not be padded with asterisks, but instead must be padded with blanks (x'40'), nulls (x'00') or underscores (x'6D'). The reason for this exception to the general masking rule, is to allow a specific search for profiles containing generic characters.

As zSecure CICS Toolkit returns the profile information into the COMMAREA, you must build the search mask in working storage and move it to the COMMAREA before each call to the API. There is also an API_RESERVED field used by the API during a search and the contents of this field must be preserved between calls.

To retrieve all profiles (for example, all user profiles), the search mask must be all asterisks. Initiate the search by specifying S in the code field. You can then retrieve the rest of the profiles by setting the code field to N. Continue to call the API until a non-zero return code is returned. The API_MSG field also contains a message indicating that there are no more profiles matching the criteria or that you have reached the end of the RACF database. Remember to recreate the search mask before each call.

Implementing field or record level security

Field or record level security can be implemented by using the API resource authorization checking capabilities, especially when using resource names of up to 246 bytes.

About this task

By defining resource names that represent a particular field or record within a file, access to those records/fields can be restricted. An application program can call the
API to verify the access authority of a user and determine what action must be taken. The action includes updating the record, displaying the record, updating the field, blanking out the field, and so on.

An example might be for a file with a DDNAME of PAYFILE, whose keys are social security numbers.

**Procedure**

1. Define the DDNAME and social security numbers to RACF:
   
   ```
   RDEFINE RSRCLASS PAYFILE.999-99-999 UACC(NONE)
   ```

2. Permit the user to the record:
   
   ```
   PERMIT PAYFILE.999-99-999 CLASS(RSRCLASS) ID(USERIDA) ACC(READ)
   ```

**What to do next**

The application can now call the API to perform a resource access check and determine the users' level of access to the record/field.

### Access Authority Check function

You can use the Access Authority Check function to determine if a user has access to one or more resources. No authority is required.

**COMMAREA**

Minimum size 99 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library. Example:

- `API_FUNC DC CL4'RSRC' FUNCTION code for access check`
- `API_RC DC XL01'00' Return code`
- `API_MSG DC CL79' ' Message area`
- `API_RSRC_NAMES DC XCL14 This is a list of resources for which the access authority of the user signed on at the terminal is to be checked. The size of this field depends on the number of resources being checked. Each resource name requires thirteen bytes, padded with blanks, followed by a one byte return code field.`
- `The return code field may also specify the level access to be checked. This may be 'R' (read), 'U' (update), 'C' (control) or 'A' (alter). Read is the default.`
- `For example to check the users access to AUDIT and PAYROLL the following entries could be coded.`
- `DC CL13'AUDIT' First resource name`
- `DC XL1'00' Return Code`
- `DC CL13'PAYROLL' Next resource name`
- `DC XL1'00' Return Code`
- `DC XL1'FF' The last field in the COMMAREA must be a one byte field containing X'FF', indicating the end of the list of resource names.`
The resource class used when making the access check is that specified in the RSRCLASS parameter in CQTPCNTL.

If a prefix has been specified for application resource names (see the CICSAPPL parameter of CQTPCNTL), it is used to prefix the resource names passed to CQTPAPI0. Refer to Chapter 4, “Application security management,” on page 29 for more information about application security and defining resources to RACF.

SMF records are produced depending on the AUDIT parameters specified for the resources. If you want to suppress the ICH408I messages on the system console and the CICS log, you can specify the value S in the API_RC field. Specifying the value results in suppression of possible access violation messages, while still creating SMF records about these violations.

The return codes are a 1-byte hexadecimal field with the following meanings:

<table>
<thead>
<tr>
<th>RETURN CODE</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'00'</td>
<td>Access allowed to resource.</td>
</tr>
<tr>
<td>X'04'</td>
<td>The resource is not defined to RACF.</td>
</tr>
<tr>
<td>X'08'</td>
<td>The user is not authorized to use the resource.</td>
</tr>
<tr>
<td>X'0C'</td>
<td>RACF is not active.</td>
</tr>
<tr>
<td>X'10'</td>
<td>FRACHECK installation exit error.</td>
</tr>
<tr>
<td>X'14'</td>
<td>RACF is not installed or at the wrong level.</td>
</tr>
</tbody>
</table>

**Access Authority Check (Extended) function**

You can use the Access Authority Check (Extended) function to check if a user has access to one or more resources. No authority is required.

**COMMAREA**

The minimum size is 348 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library. Example:

```assembly
API_FUNC DC CL4'RSRX' Function code for access check
API_RC   DC XL01'00' Return code
API_MSG  DC CL79' ' Message area
  * API_RSRX_USERID DC CL8 Specify the USERID to be used to perform third party authorization checking.
  * If blank the ACEE of the user Signed on at the terminal is used.
  *
  * API_RSRX_CLASS DC CL8 This field may be used to specify the resource class to be used.
  * The access checks, if used, it overrides the definition x for RSRCLASS in CQTPCNTL.
  * It must be x a valid class defined in the SIT unless a userid has been specified in API_RSRX_USERID. In this case any class may be specified.
  *
  * API_RSRX_NAMES DC XCL247 This is a list of resources for which the access authority of the user signed on at the terminal is to be checked.
  * The size of this field depends on the number of resources being checked. Each resource name requires 246 bytes, padded with blanks, followed by a one byte return code field.
  *
  * The return code field may also specify the level of access to be checked.
```
The access level

For example to check the users access to AUDIT and PAYROLL the following entries could be coded.

```
DC CL246'AUDIT' First resource name
DC XL1'00' Return Code
DC CL246'PAYROLL' Next resource name
DC XL1'00' Return Code
```

The last field in the COMMAREA must be a one byte field containing X'FF', indicating the end of the list of resource names.

The resource class used when making the access check is that specified in the RSRCLASS parameter in CQTPCNTL. If specified, the value in API_RSRX_RSRCLASS overrides that in CQTPCNTL.

If a prefix has been specified for application resource names (see the CICSAPPL parameter of CQTPCNTL), it is used to prefix the resource names passed to CQTPAPI0. Refer to chapter 4, “Application security management,” on page 29 for more information about application security and defining resources to RACF.

SMF records are produced depending on the AUDIT parameters specified for the resources. If you want to suppress the ICH408I messages on the system console and the CICS log, you can specify the value N in the API_RC field. Specifying the value results in Suppression of possible access violation messages, while still creating SMF records about these violations.

The return codes are a 1-byte hexadecimal field with the following meanings:

**RETURN CODE:**

- `X'00'` Access allowed to resource.
- `X'04'` The resource is not defined to RACF.
- `X'08'` The user is not authorized to use the resource.
- `X'0C'` RACF is not active.
- `X'10'` FRACHECK installation exit error.
- `X'14'` RACF is not installed or at the wrong level.

### Resource Profile List function

You can use the Resource Profile List function to provide a list of the profiles accessible to a user. No authorization is required.

You can use a function provided through the zSecure CICS Toolkit API for high-performance listing of all authorized profiles in a specified resource class. This API provides an alternative to the combined SEARCH and RLIST interfaces. You can use it to list all profiles in the specified resource class to which a user has access at a certain level. Internally, the API is based on high-performance RACF functions like the Profile Name List function (IRRPNL00) and the fast authorization checking function (RACROUTE REQUEST=FASTAUTH). This function is available only through the API.
COMMAREA
Minimum size 146 bytes.

In your application, use the APICOMMA or APICOMMC mapping macros (copybooks) provided in the SCQTMAC library.

API_FUNC DC CL4'RSRL' FUNCTION code for Resource List
API_RC DC XL1'00' API Return code
API_MSG DC CL9' ' Message area

* API_RSRL_RETC DS XL1 Return code
API_RSRL_REAS DS XL1 Reason code
API_RSRL_CLASS DS CL8 CLASS
API_RSRL_USERID DS CL8 USERID or blank
API_RSRL_GROUP DS CL8 GROUPID or blank
API_RSRL_TSQUEUE DS CL16 TSQUEUE name or blank
API_RSRL_Prefix DS CL16 Filter prefix or blank
API_RSRL_Retflag DS CL1 Processing flags
*   "C" Return data in Commarea
*   "T" Return data in TSQUEUE
*   "B" Return data in Comm/TSQ
*   "N" Don't return any data
API_RSRL_FL_PR DS CL1 Use filter prefix Y/N
API_RSRL_FL_AC DS CL1 Return all profiles Y/N
API_RSRL_ACCESS DS CL1 Requested Access
*   "R" Read
*   "U" Update
*   "C" Control
*   "A" Alter
API_RSRL_PROFCNT DS XL4 Returned number of entries
API_RSRL_PROFSLST DS XL2 Length of the profile below (CLx)
DS CL1 Flag byte
* Possible values for the flag byte
*   "A" Access to discrete profile
*   "N" READ to discrete profile
*   "B" Access to generic profile
*   "G" READ to generic profile
DS CLx Profile

Output of this function consists of a list of profile names. The list is returned in the provided commarea or in the specified Temporary Storage Queue (TSQUEUE). The return option set in API_RSRL_RETFLAG determines which of the return areas is used. The list of profiles contains either all profiles to which the user has at least READ access or only those profiles to which the user has at least the access specified in API_RSRL_ACCESS. The API_RSRL_FL_AC flag byte controls what the list of profiles contains. You can filter the list of profiles based on the first characters of the profile name. Specify the filter pattern in the field API_RSRL_Prefix. Filtering is activated by the API_RSRL_FL_PR flag byte.

When profiles are to be returned in a CICS TSQUEUE, a non-recoverable MAIN-storage TSQUEUE must be specified. When using a recoverable TSQUEUE, the program might terminate with an ATSP ABEND. Using an AUXILIARY storage queue might involve I/O to auxiliary storage, impacting the application response time. If you want the associated system resources freed, the calling program must delete the TSQUEUE after processing the data.

When profiles are to be returned in the provided COMMAREA, be sure that the commarea is sufficiently large. If the output data does not fit in the provided space, the data is truncated and a message is issued. This RSRL API does not allow retrieval of the remainder of the truncated data. If additional data is required, the request must be re-issued providing a larger commarea, or requesting that output is returned in a TSQUEUE.
The specified resource class must have been RAClISTed using the RACF SETROPTS command. Globally RAClISTed resource classes, like the CICS resource class TCICSTRN, are not supported.

The following list describes the fields in the API commarea.

**API_FUNC**
- Describes the function being called. For the List Authorized Resources function, it must contain ‘RSRL’.

**API_RC**
- The API return code. This is the return code for the API interface. The return code from the RSRL function is provided in the API_RSRL_RETC field. Possible values for the API_RETC are described in “Command requests using the COMMAREA” on page 87.

**API_MSG**
- A warning or error message.

**API_RSRL_RETC**
- The return code from the RSRL function. See “Return and Reason codes” on page 96 for a description of the possible return codes.

**API_RSRL_REAS**
- The reason code from the RSRL function. See “Return and Reason codes” on page 96 for a description of the possible reason codes.

**API_RSRL_CLASS**
- The resource class for which the authorized profiles are required. This resource class must be RAClISTed using the SETROPTS RAClIST command. Globally RAClISTed resource classes, like the CICS resource class TCICSTRN, are not supported. If the resource class is not SETROPTS RAClIST, an error message is issued and execution stops. The class name must be specified in full. Abbreviations and generic characters are not supported.

**API_RSRL_USERID**
- The user for which the list of authorized profiles is to be determined. If this field is empty, the list of authorized profiles is determined for the logged-on terminal user. The field is considered empty if the first position contains either a blank or a hexadecimal null.

**API_RSRL_GROUP**
- The RACF group to be used for the user specified in API_RSRL_USERID. This field must either be empty or contain a valid non-revoked group connect for the specified user. This field is ignored if the API_RSRL_USERID field is empty.

**API_RSRL_TSQUEUE**
- Specifies the CICS Temporary Storage Queue (TSQUEUE) to be used for the output data. The TSQUEUE name is up to 16 characters long, and must be padded with blanks or nulls. The name must be that of a non-recoverable MAIN-storage TSQUEUE. Recoverable TSQUEUEs are not supported. Using an AUXILIARY storage queue might involve I/O to auxiliary storage, impacting the application response time. If you want the associated system resources freed, the calling program must delete the TSQUEUE after processing the data. This field is ignored if the API_RSRL_RETFLAG has any value other than T or B.

**API_RSRL_PREFIX**
- Specifies the prefix filter to be used for filtering the authorized profiles.
The prefix is up to 16 characters long, and must be padded with blanks or nulls. The prefix is used to compare against the profiles, similar to the process used for the TSO SEARCH MASK keyword. The first characters of the profile name must match the characters specified for the prefix filter. If these characters do not match, the profile is skipped and is excluded from the list of profiles.

**API_RSRL_RETFLAG**

Specifies processing flags. The following values for the processing flags can be used:

- **C**  The output data is returned in the COMMAREA. The number of entries located is shown in the API_RSRL_PROFCNT field. If the COMMAREA is too small, this number can be higher than the actual number of profiles in the COMMAREA.

- **T**  The output data is returned in the specified TSQUEUE. The number of entries located is shown in the API_RSRL_PROFCNT field.

- **B**  The output data is returned in the COMMAREA and the specified TSQUEUE. The number of entries located is shown in the API_RSRL_PROFCNT field.

- **N**  No output data is returned in either the COMMAREA or the specified TSQUEUE. The number of entries that might have been returned is still shown in the API_RSRL_PROFCNT field.

**API_RSRL_FL_PR**

The prefix filter is used to limit the profiles listed. Possible values are Y or N. Any other value is treated as if the value is N.

**API_RSRL_FL_AC**

The output data always includes only those profiles to which the user has at least READ access. Profiles to which the user has no access are never shown. The API_RSRL_FL_AC flag can be used to reduce the profiles returned. Possible values for the flag are Y and N. Any other value is treated as if the value is N.

If this flag has the value N, only the profiles to which the user has at least the requested access are returned. If this flag has the value Y, all profiles to which the used has at least READ access are returned, and the profile flag shows whether the user has only READ access or the requested access.

**API_RSRL_ACCESS**

Specifies the minimum required access of the user. Possible values are R for READ, U for UPDATE, C for CONTROL, or A for ALTER. If this field is empty or contains any other value, access is determined as if the value R was specified. If the user has access at the requested access level or higher, the profile flag has the value A or B. If the user does not have the requested access, the profile flag has the value N or G.

**API_RSRL_PROFCNT**

This output field shows the number of entries listed. If the response area is large enough, it contains the number of profiles returned. If the response area is too small, it contains the number of profiles that would have been returned if the response area were large enough.

**API_RSRL_PROFLST**

This part of the commarea contains the list of authorized profiles. It consists of an array of profiles using the following format:
NAME LENGTH
The 2-byte length of the profile name. This value does not include
the length of the length field itself or the length of the FLAG byte.

FLAG A 1-byte flag field. Possible values for this return flag are:
A The user has access to the discrete profile at the requested
access level or higher.
B The user has access to the generic profile at the requested
access level or higher.
N The user does not have the requested access to the discrete
profile. The user has READ access to the profile.
G The user does not have the requested access to the generic
profile. The user has READ access to the profile.

PROFILE NAME
A variable-length profile name

If the application requests to return those profiles to which the user has at least
READ access, the profile flag for the returned profiles has the value A or B. In this
situation the values N and G are not used. If the requested access is one of the other
values, all four values for the profile flag are used.

TSQUEUE usage for profiles
If the application specifies the value T or B in the API_RSRL_RETFLAG field, the
profiles are returned in the TSQUEUE specified in the API_RSRL_TSQUEUE field.

The TSQUEUE must be a non-recoverable MAIN-storage TSQUEUE. Recoverable
TSQUEUEs are not supported. Using an AUXILIARY storage queue is discouraged
because it might involve I/O to auxiliary storage, impacting the application
response time. The API program clears the entire TSQUEUE before writing any
records. If you want the associated system resources freed, the calling program
must delete the TSQUEUE after processing the data.

Each of the requested profiles is written in a separate record. The layout of the
record is identical to that of the API_RSRL_PROFLST shown in the preceding list.

Return and Reason codes
Some of the return and reason codes returned by this API are specific ones for this
function, and some are the ones used by RACF for the IRRPNL00 function.

The following list summarizes the specific return and reason codes. See z/OS
Security Server RACF Macros and Interfaces for information about the return and
reason codes for the IRRPNL00 function.

RC=00 No error occurred. The requested profiles are provided in the specified
areas.
RC=04 See IRRPNL00 return and reason codes.
RC=08 See IRRPNL00 return and reason codes.
RC=0C

REAS=00 The internal work area used to process the authorized profiles
is too small. Only profile list requests that need less than 128Kbyte of
data can be processed.
REAS=04 Profile return in a TSQUEUE was requested, but no TSQUEUE name is given.
REAS=08 The terminal user does not have access to the specified TSQUEUE name.
REAS=0C The COMMAREA provided is not large enough to contain all authorized profiles.

RC=14 to RC=24
See IRRPNL00 return and reason codes.

RC=32 RACROUTE REQUEST=VERIFY for the specified user failed. See message CQT030 for the RACF return and reason codes.

### ADDGROUP / ALTGROUP / DELGROUP function (add, alter, or delete a group)

Use the ADDGROUP, ALTGROUP, and DELGROUP function to add a new group to the system or to alter or delete an existing group.

#### AUTHORITY
The user must have access to the zSecure CICS Toolkit command (TOOLKIT.ADGR / TOOLKIT.ALGR / TOOLKIT.DELG / TOOLKIT.LGRP, depending in the command being performed) and the group (ADGR.<grpname> / ALGR.<grpname> / DELG.<grpname> / LGRP.<grpname>).

#### COMMAREA
Minimum size for this function is 370 bytes in order to support Universal groups.

In your application, use the CQTMPIA or CQTMPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC DC CL4'ADGR' Function code
API_RC   DC XL01'00'  Return code
API_MSG  DC CL79' ' Message area
  *
API_AGRP_RC DC XL1 Return code from requested command
  * If non-zero the command failed.
  * API_MSG will give the reason for the failure.
  *
API_AGRP_CODE1 DC CL4"xxxx" 'ADGR' for ADDGROUP
  * 'ALGR' for ALTGROUP
  * 'DELG' for DELGROUP
  * 'LGRP' for LISTGROUP

*API_AGRP_GROUP DC CL8 Group name.
*API_AGRP_OWNER DC CL8 Owner name
*API_AGRP_SUPGRUP DC CL8 Superior Group name
*API_AGRP_TERMUAC DC CL1 Terminal UACC ('Y' or 'N')
*API_AGRP_INSTDATA DC CL255 Installation data
*API_AGRP_UNIVER DC CL1 Universal Group ('Y' or 'N')
```

To retrieve information about the group, enter LGRP in the API_AGRP_CODE1 field. When altering data, blank fields are ignored and are not updated. To DELETE the installation data field, set the first byte to binary zeros (X'00').
ADDUSER function (add user profile)

Use the ADDUSER function to add a new user profile to the system.

AUTHORITY

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.ADUS) and the default group of the user being added (ADUS.dfltgrp).

COMMAREA

Minimum size 408 bytes.

If your application reserves space for the no longer supported automatic create of a CICS segment, the required size would be 495 bytes.

If you need to specify a password phrase, the required size is 595 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

API_FUNC DC CL4'ADUS' Function code for ADDUSER
API_RC DC XL01'00' Return code
API_MSG DC CL79'' Message area
* 
API_ADUS_RC DC XL1 Return code from ADDUSER.
* 
API_ADUS_USERID DC CL8 Userid being added.
* 
API_ADUS_PGMNAME DC CL20 Users name.
* 
API_ADUS_DFLTGRP DC CL8 The users default group.
* 
API_ADUS_AUTHRTY DC CL1 Authority in the default group.
* 
API_ADUS_SMTWTFS DC CL7 The days of the week the user can logon.
* 
API_ADUS_FROM DC CL4 The time of day the user can logon from (24 hour clock).
* 
API_ADUS_TILL DC CL4 The time of day the user can logon till (24 hour clock).
* 
API_ADUS_INSTDATA DC CL255 Installation data field.
* 
API_ADUS_PASSWORD DC CL8 Initial password for the user.
* 
API_ADUS_OWNER DC CL8 The owner of the profile.
* 
* THE FOLLOWING FIELDS ARE USED FOR AUTOMATIC ADD
* OF CICS SEGMENT. THIS IS NO LONGER SUPPORTED. THE FIELDS
* SHOULD BE BLANKS OR NULLS.
* 
API_ADUS_OPIDENT DC CL3 Retained for compatibility
API_ADUS_OPPRTY DC CL3 Retained for compatibility
API_ADUS_TIMEOUT DC CL3 Retained for compatibility
API_ADUS_XRFSOFF DC CL7 Retained for compatibility
If you do not specify a value when creating the user profile, the initial PASSWORD for the user is set to the value of the DEFAULT GROUP.

The first time users log on, they are required to enter a new password.

**ALTUSER function (changing a profile)**

Use the ALTUSER function to change the profile for a specific user.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the user's default group (AUSR.dfltgrp).

For several fields, the user must also have system special, or access to TOOLKIT.SPEC. These fields are flagged with an asterisk (*).

**COMMAREA**

Minimum size 487 bytes.

If you need to specify a password phrase, the required size is 587 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC DC CL4'AUSR' Function code for ALTUSER
API_RC DC XL01'00' Return code
API_MSG DC CL79' ' Message area
* 
API_ALUS_RC DC XL1 Return code from ALTUSER
* If non-zero the command failed.
* API_MSG will give the reason for
* the failure.
* 
API_ALUS_USERID DC CL8 The userid to be altered.
API_ALUS_PASSWORD DC CL8 Password
API_ALUS_RESUME DC CL1 Resume the userid (Y or N)
API_ALUS_PGMRNAME DC CL20 Name
API_ALUS_INSTDATA DC CL255 Installation data field.
API_ALUS_DFLTGRP DC CL8 Default group.
API_ALUS_REVOKEDC DC CL5 Revoke Date(YYDDD).
API_ALUS_RESUMED DC CL5 Resume Date(YYDDD).
API_ALUS.Author DC CL8 * OWNER
API_ALUS_GRPACC DC CL1 * Group access.
API_ALUS_ADSP DC CL1 * ADSP.
API_ALUS_SPEC DC CL1 * Special
API_ALUS_OPER DC CL1 * Operations
API_ALUS_AUDITOR DC CL1 * Auditor
API_ALUS_RESTR DC CL1 * UACC and similar not used.
API_ALUS_PROTECT DC CL1 * Password cannot be used.
API_ALUS_UAUDIT DC CL1 * Audit all RACHECK's/RACDEF's.
API_ALUS_LOGDAY DC CL7 * Days user can logon.
API_ALUS_LOGFROM DC CL4 * Starting time for logon.
API_ALUS_LOGTILL DC CL4 * Latest time for logon.
API_ALUS_MODEL DC CL44 * Dataset profile model.
API_ALUS_CLAUTH DC CL8 * Give class authority.
*API_ALUS-AUTH Name used in COBOL copybook
*API_ALUS-NAUTH Name used in COBOL copybook
API_ALUS_PASSEXPI DC CL1 * New password is expired (Y or N).
API_ALUS_PHRASE DC CL100 The password phrase of the userid.
```
The fields in the commarea must be initialized to BINARY ZEROES. Only fields that are to be altered need to contain data. Refer to Chapter 5, “The zSecure CICS Toolkit command interface,” on page 35 for a description of the fields and the restrictions on who can update which fields. After linking to CQTPAPI0, the user profile will be updated with the information contained in any field that was not binary zeros.

To specify that the password phrase must be removed, specify a value of 100 blanks. Any other value results in the password phrase being changed into the specified value, or be retained at its current value.

You can use the special date zeros (\texttt{c'00000' = x'F0F0F0F0F0'} to remove the Revoke/Resume date. Using this special value, you might implement a function like the NOREVOKE and NORESUME keywords of the z/OS 1.7 RACF \texttt{ALTUSER} command.

\textbf{ALTUSER (CICS SEGMENT) function (alter CICS segment)}

Use the ALTUSER (CICS SEGMENT) function to change the CICS segment for a specific user.

\textbf{AUTHORITY}

The user must have access to the zSecure CICS Toolkit command \texttt{TOOLKIT.AUSR} and the default group of the user (\texttt{AUSR.dflgrp}). In addition, for managing the CICS segment, the user must have access to \texttt{TOOLKIT.ACIC}. In version 1.4 of Consul zToolkit, this requirement was only enforced if the \texttt{TOOLKIT.ACIC} profile has been defined or is covered by a generic profile. In version 1.8.1 and higher of zSecure CICS Toolkit, access to resource \texttt{TOOLKIT.ACIC} is required.

\textbf{COMMAREA}

Minimum size is 184 bytes. If your application requires access to the TSLKEY and RSLKEY, the required minimum size is 316 bytes.

In your application, use the CQTMPIA or CQTMPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC DC CL4'ACIC' Function code for CICS segment.
* For compatibility reasons, ACSG is also accepted
API_RC    DC XLB1'00'  Return code
API_MSG   DC CL79' '  Message area
*  
API_ACSG_RC DC XL1   Return code from ALTUSER. If non-zero the command failed.
  * API_MSG will give the reason for the failure.
  *
API_ACSG_CODE DC CL4 Specify LIST to retrieve the current specifications for the user. See LISTUSER (TSD/CICS)
  * UPDT to update the CICS segment
  * DELT to delete the CICS segment.
API_ACSG_USERID DC CL8 The userid to be listed/updated.
API_ACSG_OPIDENT DC CL3 Three character opident
API_ACSG_OPRTY DC CL3 Operator priority (000-255)
API_ACSG_TIMEOUT DC CL3 See ALTUSER command
API_ACSG_XRFSOFF DC CL7 FORCE or NOFORCE
API_ACSG_OPCLASS DC CL71 Operator classes (01-24, separated by a comma, e.g.; 01,03,04)
*  
API_ACSG_TSLKEY DS CL66 TSL KEYS (00, 99, or 01-64, separated by commas)
```
When the CICS segment is updated, all fields in the CICS segment are replaced. For this reason, valid data must be supplied for all parameters.

**Note:** In the current release, zSecure CICS Toolkit does not verify that each TSLKEY and RSLKEY value only provides space for 22 TSLKEY and RSLKEY values.

### ALTUSER (TSO SEGMENT) function (change TSO segment)

Use the ALTUSER (TSO SEGMENT) function to change the TSO segment for a specific user.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the default group of the user (AUSR.dflgrp). In addition, for managing the TSO segment, the user must have access to TOOLKIT.ATSO. In version 1.4 of Consul zToolkit, this requirement was only enforced if the TOOLKIT.ATSO profile has been defined (or is covered by a generic profile). In version 1.8.1 and higher of zSecure CICS Toolkit, access to resource TOOLKIT.ATSO is required.

**COMMAREA**

Minimum size 191 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
* API_FUNC DC CL4'ATSO' Function code for TSO segment.
  * For compatibility reasons, ATSG is also accepted
* API_RC DC XL01'00' Return code
* API_MSG DC CL79='' Message area
  *
* API_ATSG_RC DC XL1 Return code from ALTUSER. If
  * non-zero the command failed.
  * API_MSG will give the reason for
  * the failure.
  *
* API_ATSG_CODE DC CL4 Specify LIST to retrieve the current
  * specifications for the user.
  * See LISTUSER (TSO/CICS)
  * UPDT to update the TSO segment
  * with the values in the COMMAREA.
  * DELT to delete the TSO segment.
* API_ATSG_USERID DC CL8 The userid to be listed/updated.
* API_ATSG_ACCTNUM DC CL40 The account number
* API_ATSG_DESTID DC CL8 The destination id.
* API_ATSG_HCLASS DC CL1 The hold class.
* API_ATSG_JCLASS DC CL1 The job class.
* API_ATSG_MSGCLASS DC CL1 The message class.
* API_ATSG_SCLASS DC CL1 The sysout class.
* API_ATSG_SECLABL DC CL8 The security label.
* API_ATSG_SIZE DC CL7 The region size.
* API_ATSG_MAXSIZE DC CL7 The maximum region size.
* API_ATSG_PROC DC CL8 The logon proc.
* API_ATSG_UNIT DC CL8 The allocation device.
* API_ATSG_UDATA DC CL4 The installation data.
```
When the TSO segment is updated, all fields in the TSO segment are replaced. For this reason, valid data must be supplied for all parameters. Leaving a field empty (blanks or nulls) results in the corresponding field in the TSO segment to be deleted.

**ALTUSER (OMVS SEGMENT) function (change OMVS segment)**

Use the ALTUSER (OMVS SEGMENT) function to change the OMVS segment for a specific user.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.AUSR) and the default group of the user (AUSR.dflgrp). In addition, for managing the OMVS segment, the user must have access to TOOLKIT.AOMV.

**COMMAREA**

Minimum size 273 bytes. If your applications access to the MEMLIM and SHMMAX fields, the minimum size is 291 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```plaintext
API_FUNC DC CL4'AOMV' Function code for OMVS segment.
API_RC DC XL01'00' Return code
API_MSG DC CL79' ' Message area
* _ _
API_AOMV_RC DS XL01 Return code from command
* _ _
API_AOMV_CODE DS CL04 UPDT = Update segment
* _ _
LIST = Retrieve segment
* _ _
DELT = Delete segment
  *
  *
API_AOMV_USERID DS CL08 The userid
API_AOMV_UID DS CL10 UID (1-10# Left align, AUTOUID)
API_AOMV_SHARED DS CL01 Shared UID N/Y
API_AOMV_MKDIR DS CL01 MKDIR N/Y
API_AOMV_HOME DS CL60 HOME (1-60 CHAR$) mixed case
API_AOMV_PROGRAM DS CL60 SHELL PROGRAM (1-60 CHAR$)
API_AOMV_ASSIZE DS CL10 ASSIZEMAX (1-10 Digits left a1)
API_AOMV_CPUTIME DS CL10 CPUTIMEMAX (1-10 Idem)
API_AOMV_FILEPROC DS CL06 FILEPROCMAX (1-6 Idem)
API_AOMV_MMAPAREA DS CL08 MMAPAREA (1-8 Idem)
API_AOMV_PROCUSER DS CL05 PROCUSER (1-5 Idem)
API_AOMV_THREADS DS CL06 THREADS (1-6 Idem)
API_AOMV_MEMLIM DS CL09 MEMLIMIT (1-8 DIGITS + M/G/T/P)
API_AOMV_SHMMAX DS CL09 SHMEMMAX (1-8 DIGITS + M/G/T/P)
```

When the OMVS segment is updated, all fields in the OMVS segment are replaced. For this reason, valid data should be supplied for all parameters. Leaving a field empty (blanks or nulls) results in the corresponding field in the OMVS segment to be deleted.

**ALTUSER (WORKATTR SEGMENT) function (change WORKATTR segment)**

Use the ALTUSER (WORKATTR SEGMENT) function to change the WORKATTR segment for a specific user.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command
(TOOLKIT.AUSR) and the default group of the user (AUSR.dflgrp). In addition, for managing the OMVS segment, the user must have access to TOOLKIT.AWRK.

**COMMAREA**

Minimum size 637 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC DC CL4'AWRK'   Function code for WORKATTR segment.
API_RC   DC XL01'00'    Return code
API_MSG  DC CL79' '    Message area
*        
API_AWRK_RC DS XL01     Return code from command
*        
API_AWRK_CODE DS CL04   UPDT = Update segment
*        
API_AWRK_NAME DS CL60   WANAME (1-60 Chars)
API_AWRK_ACCT DS CL60   WAACCT (1-60 Chars)
API_AWRK_BLDG DS CL60   WABLDG (1-60 Chars)
API_AWRK_DEPT DS CL60   WADEPT (1-60 Chars)
API_AWRK_ROOM DS CL60   WAROOM (1-60 Chars)
API_AWRK_ADDR1 DS CL60  WAADDR1 (1-60 Chars)
API_AWRK_ADDR2 DS CL60  WAADDR2 (1-60 Chars)
API_AWRK_ADDR3 DS CL60  WAADDR3 (1-60 Chars)
API_AWRK_ADDR4 DS CL60  WAADDR4 (1-60 Chars)
```

When the WORKATTR segment is updated, all fields in the WORKATTR segment are replaced. For this reason, valid data must be supplied for all parameters. Leaving a field empty (blanks or nulls) results in the corresponding field in the WORKATTR segment to be deleted.

**CONNECT function (connect a user or group to a group)**

Use the CONNECT function to connect a user or group to a group.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.CONN) and the target group (CONN.dflgrp)

**COMMAREA**

Minimum size 112 bytes. If your application requires access to the REVOKE and RESUME dates, the minimum length is 122 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC DC CL4'CONN'   Function code for CONNECT
API_RC   DC XL01'00'    Return code
API_MSG  DC CL79' '    Message area
*        
API_CONN_RC DS XL1      Return code from CONNECT. If non-zero the command failed.
*        
API_CONN_USERID DS CL8  Userid/group being connected.
*        
API_CONN_GROUP  DS CL8  Group being connected to.
*        
API_CONN_AUTH  DS CL1   Connect authority
*        
Must be 'U' (use),
*        
'C' (create)
*        
'N' (connect)
```
API_CONN_OWNER DC CL8  Owner of the connect profile. It must be a valid userid or group.

API_CONN_SPEC DC CL1  Specify 'Y' if the user should have the group-special attribute otherwise specify 'N'.

API_CONN_OPER DC CL1  Specify 'Y' if the user should have the group-operations attribute otherwise specify 'N'.

API_CONN_REVOKE DC CL5  The date (YYDDD) the user is to be REVOKED

API_CONN_RESUME DC CL5  The date (YYDDD) the user is to be RESUMED

The API_CONN_REVOKE and API_CONN_RESUME fields can be used to set or remove the REVOKE and RESUME dates for the connection. If you use the value blanks (x'4040404040'), the revoke and resume dates are left at their current value. If you use the special date zeros (c'00000' = x'F0F0F0F0F0'), the current Revoke/Resume date is removed. Using this last special value, you can implement a function like the NOREVOKE and NORESUME keywords of the z/OS 1.7 RACF CONNECT command. If you specify today’s date for either the REVOKEDT or the RESUMEDT, the revoke-status for the user is updated immediately and the other date value is ignored. Both the RESUMEDT and the REVOKEDT are reset.

DELETE DATASET function (delete data set profile)

Use the DELETE DATASET function to delete a data set profile from the system.

AUTHORITY

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.DELD) and the high-level-qualifier of the data set profile name (DELD.hlq). If the user does not have access to the DELD.hlq, standard RACF authority checking is used. Refer to the RACF Command Language Reference manual for information about which data set profiles a user is authorized to delete.

COMMAREA

Minimum size 130 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

API_FUNC DC CL4'DELD'  Function code for DELDSD
API_RC DC XL01'00'  Return code
API_MSG DC CL79' '  Message area
API_DELD_RC DC XL01'00'  Return code from DELDSD.
* If non-zero the command failed. API_MSG will give the reason for the failure.
*
API_DELD_DSNAME DC CL44'DS-Profile'  The dataset profile to be deleted.

API_DELD_GENERIC DC CL01'Y'  Specify 'Y' if the profile is Generic or 'N' if it is not.

DELETE USERID function (delete user profile)

Use the DELETE USERID function to delete a user ID from the system.
AUTHORITY

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.DELU) and the default group of the user ID (DELU.dfltgrp).

COMMAREA

Minimum size 93 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

API_FUNC DC CL4'DELU' Function code for DELUSER
API_RC DC XL01'00' Return code
API_MSG DC CL79' ' Message area
API_DELU_RC DC XL01'00' Return code from DELUSER.
  * If non-zero the command
  * failed. API_MSG will give the
  * reason for the failure.
  *
API_DELU_USERID DC CL8'USERID' The userid to be deleted.

The user ID must be REMOVED from all groups, except the default group, and no data set profiles using this user ID as a high-level qualifier must exist, before the DELETE is issued.

zSecure CICS Toolkit checks for group connections but not for data set profiles.

---

LISTDATASET function (list profile for one or more data sets)

Use the LISTDATASET function to list the profile for a specific data set or data sets.

AUTHORITY

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.LDSD).

COMMAREA

Minimum size 524 bytes to display the data set profile or 2374 if requesting the users or programs.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

Note: If you want to perform a SEARCH, initialize all fields according to your search pattern. Use asterisks for all field padding except for the DSETID field, which must be padded with nulls, blanks, or underscores.

API_FUNC DC CL4'LDSD' Function code for LISTDATASET
API_RC DC XL01'00' Return code
API_MSGDC DC CL79' ' Message area

  *
  *
API_LDSD_RC DC XL1 Return code from LISTDATASET.
  * If non-zero the command
  * failed. API_MSG will give the reason for
  * the failure.
  *
  *
API_LDSD_CODE1 DC CL1 Request code.
  * 'S' = start search
  * 'N' = get next profile
  * 'L' = retrieve this profile
  * 'U' = retrieve users
  * 'P' = retrieve programs
  *
API_LDSD_RESERVED DC CL46 This field is reserved for the
* API and must be preserved between calls.

* API_LDSD_DTYPE DC CL1 Profile type (generic or discrete).
  * API_LDSD_DSETID DC CL44 The dataset to be retrieved.
  * Only required when the CODE field is L, U or P; otherwise it is used as part of the search criteria.
  * API_LDSD_AUTHOR DC CL8 Owner of the profile.
  API_LDSD_CREADAT DC CL5 Creation date.
  API_LDSD_LREFDAT DC CL5 Last reference date.
  API_LDSD_LCHGDAT DC CL5 Last update date.
  API_LDSD_ACSALTR DC CL6 # of alter accesses.
  API_LDSD_ACSCNTL DC CL6 # of control accesses.
  API_LDSD_ACSUPDT DC CL6 # of update accesses.
  API_LDSD_ACSREAD DC CL6 # of read accesses.
  API_LDSD_UACC DC CL7 Universal access to the dataset.
  API_LDSD_GROPOST DC CL1 Group dataset.
  API_LDSD_AUDIT DC CL1 Audit flag.
  API_LDSD_GROUPNM DC CL8 Current connect group.
  API_LDSD_DSTYPE DC CL4 Dataset type.
  API_LDSD_LEVEL DC CL3 Level indicator.
  API_LDSD_GAUDIT DC CL1 Global audit option.
  API_LDSD_AUDITQS DC CL1 Audit success flag.
  API_LDSD_AUDITQF DC CL1 Audit failure flag.
  API_LDSD_GAUDQS DC CL1 Global audit success flag.
  API_LDSD_GAUDQF DC CL1 Global audit failure flag.
  API_LDSD_SECURITYLEVEL DC CL1 Security level.
  API_LDSD_NUMCTGY DC CL4 Number of categories.
  API_LDSD_NUMPGMS DC CL4 Number of programs.
  API_LDSD_NUMUSER DC CL4 Number of users/groups.
  API_LDSD_INSTDATA DC CL255 Installation data field.
  ORG API_LDSD_RESERVED

API_LDSD_USERPGMS DC CL?? When the users or programs are requested they will be returned into this area.

When the list of programs is returned, the format of the output is as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of program name</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Program name</td>
<td>8 bytes</td>
</tr>
<tr>
<td>Length of userid</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Userid</td>
<td>8 bytes</td>
</tr>
<tr>
<td>Length of access field</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Access</td>
<td>1 byte</td>
</tr>
<tr>
<td></td>
<td>X'80'  Alter access</td>
</tr>
<tr>
<td></td>
<td>X'40'  Control access</td>
</tr>
<tr>
<td></td>
<td>X'20'  Update access</td>
</tr>
<tr>
<td></td>
<td>X'10'  Read access</td>
</tr>
<tr>
<td></td>
<td>X'08'  Execute access</td>
</tr>
<tr>
<td></td>
<td>X'01'  None</td>
</tr>
</tbody>
</table>

When the list of users is returned, the format of the output is as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of Userid</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Userid</td>
<td>8 bytes</td>
</tr>
<tr>
<td>Length of access field</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Access</td>
<td>1 byte</td>
</tr>
<tr>
<td></td>
<td>X'80'  Alter access</td>
</tr>
<tr>
<td></td>
<td>X'40'  Control access</td>
</tr>
<tr>
<td></td>
<td>X'20'  Update access</td>
</tr>
<tr>
<td></td>
<td>X'10'  Read access</td>
</tr>
<tr>
<td></td>
<td>X'08'  Execute access</td>
</tr>
</tbody>
</table>

106 User Guide
X'01' None

Length of access count 4 bytes
Access count 2 bytes (binary)

In all cases, when the first field is zero (X'00000000'), it indicates the end of the data.

**LISTGROUP function (list profile for a group)**

Use the LISTGROUP function to list the profile for a specific group or groups.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.LGRP) and the group name (LGRP.grpname)

**COMMArea**

Minimum size 441 bytes to display the group profile or 2374 if requesting the users of subgroups. To enable Universal group support, the minimum size is 442 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

**Note:** If you want to perform a SEARCH, remember to initialize all fields according to your search pattern. Use asterisks for all field padding except for the GROUP field, which must be padded with nulls, blanks, or underscores.

<table>
<thead>
<tr>
<th>Description</th>
<th>Field</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>API_FUNC</td>
<td>DC CL4'LGRP'</td>
<td>Function code for LISTGROUP</td>
</tr>
<tr>
<td>API_RC</td>
<td>DC X'01'00'</td>
<td>Return code</td>
</tr>
<tr>
<td>API_MSG</td>
<td>DC CL79' '</td>
<td>Message area</td>
</tr>
<tr>
<td>API_LGRP_RC</td>
<td>DC X'01'</td>
<td>Return code from LISTGROUP. If non-zero the command failed.</td>
</tr>
<tr>
<td>API_LGRP_CODE1</td>
<td>DC CL1</td>
<td>Request code. 'S' = start search</td>
</tr>
<tr>
<td>API_LGRP_RESERVED</td>
<td>DC CL9</td>
<td>This field is reserved for the API and must be preserved between calls</td>
</tr>
<tr>
<td>API_LGRP_GROUP</td>
<td>DC CL8</td>
<td>The group to be retrieved. Only required when the CODE field is L, G or U otherwise it is used as part of the search criteria.</td>
</tr>
<tr>
<td>API_LGRP_SUPGRP</td>
<td>DC CL8</td>
<td>This groups superior group.</td>
</tr>
<tr>
<td>API_LGRP_OWNER</td>
<td>DC CL8</td>
<td>Owner of this group.</td>
</tr>
<tr>
<td>API_LGRP_DTE</td>
<td>DC CL5</td>
<td>Date this profile was created.</td>
</tr>
<tr>
<td>API_LGRP_UACC</td>
<td>DC CL7</td>
<td>Authority of a user to the group if the user is not connected to the group.</td>
</tr>
<tr>
<td>API_LGRP_TERMACC</td>
<td>DC CL1</td>
<td>Authority to access a terminal* required.</td>
</tr>
<tr>
<td>API_LGRP_SUBGRPS</td>
<td>DC CL5</td>
<td>Number of subgroups.</td>
</tr>
</tbody>
</table>

Chapter 7. Application programming interface (API) 107
**API_LGRP_USERS** DC CL5  Number of users.

**API_LGRP_MODEL** DC CL44  Name of a profile to be used as model for new group-name datasets.

**API_LGRP_INSTDATA** DC CL255  Installation data field.

**API_LGRP_UNIVERS** DC CL1  Universal Group ('Y' or 'N')

**API_LGRP_USERSUBG** DC CL???  When the users or subgroups are requested they will be returned into this area.

When the list of users or subgroups is returned, the format of the output is as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of member</td>
<td>4 bytes</td>
</tr>
<tr>
<td>User/Subgroup name</td>
<td>8 bytes</td>
</tr>
</tbody>
</table>

In all cases, when the first length field is zero (x'00000000'), this indicates the end of the data.

---

**LISTUSER function (list profile for a user ID)**

Use the LISTUSER function to list the profile for a specific user or user IDs.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.LUSR) and the users default group (LUSR.dfltgrp).

**COMMAREA**

Minimum size 544 bytes to display the group profile or 2374 if requesting the groups or categories.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

**Note:** If you want to perform a SEARCH, you need to initialize all fields according to your search pattern. Use asterisks for all field padding except for the USERID field, which must be padded with nulls, blanks, or underscores.

**API_FUNC** DC CL4 'LUSR'  Function code for LISTUSER.

**API_RC** DC XL01 '00'  Return code

**API_MSG** DC CL79 ''  Message area

**API_LUSR_RC** DC XL1  Return code from LISTUSER.

**API_LUSR_CODE1** DC CL1  Request code.

'S' = start search

'N' = get next profile

'L' = retrieve this profile

'G' = retrieve groups

'A' = retrieve groups and authority to groups

'C' = retrieve categories

**API_LUSR_RESERVED** DC CL9  This field is reserved for the API and must be preserved between
* calls.

API_LUSR_USERID DC CL8 The userid to be retrieved.
* Only required when the CODE field
* is L, G or C otherwise it is used
* as part of the search criteria.

API_LUSR_PGMNAME DC CL20 Users name.
API_LUSR_AUTHOR DC CL8 Owner of the profile.
API_LUSR_PASSWRD DC CL8 Password field (will contain ?)
API_LUSR_AUTHDTE DC CL5 Creation date.
API_LUSR_DFLTGRP DC CL8 Default group.
API_LUSR_AUTHRTY DC CL7 Authority.
API_LUSR_UACC DC CL7 Universal access.
API_LUSR_CLASCNT DC CL5 Number of classes.
API_LUSR_ADSP DC CL1 ADSP.
API_LUSR_SPEC DC CL1 Special.
API_LUSR_OPER DC CL1 Operations.
API_LUSR_REVOKE DC CL1 Revoke.
API_LUSR_GRPACC DC CL1 GRPACC.
API_LUSR_AUDITOR DC CL1 Auditor.
API_LUSR_PROTECT DC CL1 Password cannot be used.
API_LUSR_RESTR DC CL1 UACC and similar not used.
API_LUSR_UAUDIT DC CL1 Audit all RACHECK's/RACDEF's.
API_LUSR_REVOKEC DC CL2 # unsuccessful pwd attempts.
API_LUSR_REVOKED DC CL5 Date user will be revoked.
API_LUSR_SECL DC CL2 Security level. This is a binary field that
represents the security level. Eg.:
* X'00FE' would be a security level of 254.
* API_LUSR_RESUMED DC CL5 Date user will be resumed.
API_LUSR_LASTACC DC CL14 Last access date and time.
API_LUSR_PASSDTE DC CL5 Date password last changed.
API_LUSR_PWDINT DC CL3 Password interval.
API_LUSR_PWCDNT DC CL3 Current password generation #.
API_LUSR_NUMCTGY DC CL4 Number of categories.
API_LUSR_NUMGRP DC CL4 Number of groups.
API_LUSR_LOGDAY DC CL7 Days user can logon.
API_LUSR_LOGFROM DC CL4 Starting time for logon.
API_LUSR_LOGTILL DC CL4 Latest time for logon.
API_LUSR_MODEL DC CL44 Dataset profile model.
API_LUSR_INSTDATA DC CL255 Installation data field.

When the list of groups is returned, the format of the output is as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of group name</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Group name</td>
<td>8 bytes</td>
</tr>
</tbody>
</table>

When the list of groups and authority is returned, the format of the output is as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Combined length of the following 'group length/names'</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Length of group name</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Group name</td>
<td>8 bytes</td>
</tr>
</tbody>
</table>

For each of the following fields, there is a one-to-one relationship to the groups. If
the user was connected to two groups, there are two ADSP flags, two SPECIAL

Chapter 7. Application programming interface (API)  109
flags, two OPERATIONS flags, two REVOKE flags, two GRPACC flags, and two GROUP AUDITOR flags. If bit 0 in the flag is turned on (X'80'), the user has that attribute in the group.

Combined length of the ADSP lengths/flags 4 bytes
Length of the ADSP flag 4 bytes
ADSP flag 1 byte

Combined length of the SPECIAL lengths/flags 4 bytes
Length of SPECIAL flag 4 bytes
SPECIAL flag 1 byte

Combined length of the OPERATIONS lengths/flags 4 bytes
Length of OPERATIONS flag 4 bytes
OPERATIONS flag 1 byte

Combined length of the REVOKE lengths/flags 4 bytes
Length of the REVOKE flag 4 bytes
REVOKE REVOKE flag 1 byte

Combined length of the GRPACC lengths/flags 4 bytes
Length of GRPACC flag 4 bytes
GRPACC flag 1 byte

Combined length of the GROUP AUDITOR lengths/flags 4 bytes
Length of GROUP AUDITOR flag 4 bytes
GROUP AUDITOR flag 1 byte

When the list of categories is returned, the format of the output is as follows:

<table>
<thead>
<tr>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of category</td>
<td>4 bytes</td>
</tr>
<tr>
<td>Category number</td>
<td>2 bytes (binary)</td>
</tr>
</tbody>
</table>

In all cases, when the first length field is zero (x'00000000'), it indicates the end of the data.

**PASSWORD function (change password)**

Use the PASSWORD function to change the password of a user.

**AUTHORITY**

NONE, unless you are specifying a value of 255 for the interval value (which corresponds to NOINTERVAL), or changing the INTERVAL value for a user ID other than your own. It requires you to have SPECIAL, or to have access to TOOLKIT.SPEC or PSWD.dflgrp (dflgrp is the default group of the user ID that is being altered). If you are using the PASSWORD command on a user ID different from your own, you might only change the INTERVAL value. To change another user's password, use the ALTUSER command.

**COMMAREA**

Minimum size 112 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.
PASSWORD does not perform a signon for the user. It only verifies that the password entered for this user ID is correct and then changes the password to the new specification and changes the password interval.

Multiple failures to change the password of a user might result in the user ID being revoked, depending on your system parameters.

The password interval of the user might also be changed. The new interval might be 001 - 254, but might not exceed the global maximum specified. If it does, it can be set to the maximum allowed. If the value specified for the interval is invalid, the parameter is ignored. When only changing the password interval, it is not necessary to provide the password. However, if the interval is invalid and ignored, zSecure CICS Toolkit treats the operation as if it is a request to change passwords and it checks for a password and new password. If a new password has not been supplied, the error message returned reflect it as the error, rather than the password interval being incorrect.

This command is only available through the API.

**PERMIT function (grant or remove access)**

Use the PERMIT function to grant access to or remove access from a CICS resource. The resource must be in one of the resource classes defined in the SIT for this run of CICS.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.PEMT), the default group of the user ID or group that is being granted access (PEMT.dfltgrp), and must also have access to the resource. If the access is granted to a group, the resource used is PEMT.group. After the PERMIT has been completed, the resource classes must be refreshed in order to have immediate effect. Use the RACF `SETROPTS REFRESH` command to accomplish it.

**COMMAREA**

Minimum size 116 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.
PERMITX function (grant or remove access - any resource)

Use the PERMITX function to grant access to or remove access from any resource. It might also be used to grant access to DATASET profiles.

AUTHORITY:
The user must have access to the zSecure CICS Toolkit command (TOOLKIT.PEMT). The default group of the user ID or group that is being granted access (PEMT.dfltgrp), and must also have access to the resource. If the access is granted to a group, the resource used is PEMT.group. The user must also be authorized to the PEMX.classname profile. After the PERMIT has been completed, the resource classes must be refreshed in order to have immediate effect. Use the RACF SETROPTS REFRESH command to accomplish it.

In order for the PERMITX function to be available, it must have been enabled in CQTPCNTL by specifying PEMTALL=Y.

COMMAREA
Minimum size 349 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
* If non-zero the command failed
* API_MSG will give the reason for the failure
*
API_PEMT_USERID DC CL08'USERID' The userid or group.
*
API_PEMT_RSRC DC CL13'CEMT ' The name of the CICS resource.
*
API_PEMT_CLASS DC CL08'TCICSTRN'
  * The resource classname.
  * If blank the value of the XTRAN parameter specified in the SIT is used.
*
API_PEMT_DELT DC CL01'Y' Specify 'Y' in this field to remove
  * a person from the access list for
  * this resource. The user or group
  * will no longer have access to the resource.
*
API_PEMT_ACC DC CL01'R' Access allowed to the resource
  * Specify 'R' for read,
  * 'N' for none,
  * 'U' for update,
  * 'A' for alter or
  * 'C' for control.
  * Read is the default
```
**RACLINK function (define, list, undefine, or approve user associations)**

Use the RACLINK function to list, define, approve, and undefine RRSF user ID associations on the local system.

**AUTHORITY**

The user must have access to the zSecure CICS Toolkit command (TOOLKIT.RACL) and the default group of the user ID (PEMT.dfltgrp) and for the DEFINE function must also have access to the RACLINK.DEFINE.nodename and RACLINK.PWSYNC.nodename profiles in the RRSFDATA resource class.

**COMMAREA**

Minimum size 1150 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC DC CL4'RACL' Function code for RACLINK
API_RC DC XL01'00' Return code
API_MSG DC CL79' ' Message area
*    
API_RACL_RC DS XL01 Return code from command
API_RACL_CODE1 DS XL01 'D' DEFINE
  'U' UNDEFINE
  'A' APPROVE
  'L' LIST ASSOCIATIONS
  
API_RACL_USERID DC CL8'userid' Userid on whose behalf
API_RACL_ATYPE DC CL8'PEER' Assoc. type (PEER/MANAGED)
API_RACL_ANODE DC CL8'nodename' Assoc. node
API_RACL_AUSERID DC CL8'ibmuser' Assoc. UserID
API_RACL_PWSYNC DC CL4'yes' 'yes/no
API_RACL_APSWD DC CL8'sys1' Assoc. UserID Password
API_RACL_ASSOCLST DS 15CL68 List of 15 associations
  
The list of associations for the user has the following format:

API_RACL_ASSOCTYPE DC CL10 PEER/MANAG
API_RACL_ASSOCNODER DC CL8 node
API_RACL_ASSOCUSER DC CL8 USER
API_RACL_ASSOCPWSYNC DC CL4 pwsync
API_RACL_ASSOCSTAT DC CL20 status
API_RACL_ASSOCCREAT DC CL8 creator
API_RACL_ASSOCDATE DC CL10 date
```
REMOVE function (remove user IDs or groups from a group)

Use the REMOVE function to remove a user or group from group.

AUTHORITY
The user must have access to the zSecure CICS Toolkit command (TOOLKIT.REMV) and the target group (REMV.grpname)

COMMAREA
Minimum size 101 bytes.
In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC   DC  CL4'REMV'  Function code for REMOVE
API_RC     DC  XL01'90'   Return code
API_MSG    DC  CL79' '   Message area

*   API_REMV_RC  DC  XL1   Return code from REMOVE.
*   If non-zero the command failed.
*   API_MSG will give the reason for the failure.

*   API_REMV_USERID  DC  CL8   Userid/group being removed.
*   API_REMV_GROUP   DC  CL8   Group being removed from.
```

Users might not be removed from their default group.

RALTER/RDEFINE/RDELETE function (list and maintain profiles)

Use the RALTER, RDEFINE, and RDELETE function to list and maintain profiles in a general resource class defined in the CDT.

AUTHORITY
The user must have access to the zSecure CICS Toolkit command (TOOLKIT.RALT / TOOLKIT.RDEF / TOOLKIT.RDEL depending on the command being performed) and the general resource class (RALT.cdtclass / RDEF.cdtclass / RDEL.cdtclass / RLST.cdtclass)

COMMAREA
Minimum size 875 bytes.
In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC   DC  CL4'Rxxx'  Function code:
*   RALT for RALTER
*   RDEF for RDEFINE
*   RDEL for RDELETE
API_RC     DC  XL01'90'   Return code
API_MSG    DC  CL79' '   Message area

*   API_RUPD_RC  DC  XL1   Return code.
*   If non-zero the command failed.
*   API_MSG will give the reason for the failure.

*   API_RUPD_CODE1  DC  CL4   Type of command being performed
*   'RDEF' to define a profile
*   'RDEL' to delete a profile
*   'AMEM' to add a member
*   'DMEM' to delete a member
*   'UPDP' to update fields in the profile

*   API_RUPD_CLASS  DC  CL8   The class containing the profile
```
RLIST function (list profiles for general resource class)

Use the RLIST function to list the profile details for a general resource class defined in the CDT.

AUTHORITY
The user must have access to the zSecure CICS Toolkit command (TOOLKIT.RLST) and the general resource class (RLIST.cdtclass).

COMMAREA
Minimum size 907 bytes to display the profile. If the request is for the members, users or condacc, the commarea must be large enough to hold all the data returned. If it is not, the API_RLST_RC is non-zero and the message indicates it as the error.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

Note: If you want to perform a SEARCH, you need to initialize all fields according to your search pattern. Use asterisks for all field padding except for the ENTRY field, which must be padded with nulls, blanks, or underscores.
* search criteria.

API_RLST.Owner DC CL8 Owner of the profile.
API_RLST.DEFDATE DC CL5 Creation date.
API_RLST.LREFDAT DC CL5 Last reference date.
API_RLST.LCHGDAT DC CL5 Last update date.
API_RLST.UACC DC CL7 Universal access to the dataset.
API_RLST.AUDIT DC CL1 Audit flag.
API_RLST.AUDITQS DC CL1 Audit success flag.
API_RLST.AUDITQF DC CL1 Audit failure flag.
API_RLST.NOTIFY DC CL8 User to be notified.
API_RLST.WARNING DC CL1 Warning mode.
API_RLST_LEVEL DC CL3 Level indicator.
API_RLST.GAUDIT DC CL1 Global audit option.
API_RLST.GAUDQS DC CL1 Global audit success flag.
API_RLST.GAUDQF DC CL1 Global audit failure flag.
API_RLST_SECLEVEL DC CL3 Security level.
API_RLST_NUMMEM DC CL4 Number of members.
API_RLST_NUMUSER DC CL4 Number of users/groups.
API_RLST_NUMPGMS DC CL4 Number of programs.
API_RLST_INSTDATA DC CL255 Installation data field.

When a profile or member name is returned, it might be generic. It cannot be converted to a displayable format. To do so, you must be aware of the RACF naming conventions for generic characters.

Generic Character Converted To

The first ',' X'02'
Ending double asterisks X'FD'
Ending single asterisks X'FC'
Internal double asterisks X'FBFC90' (when a general resource class)
X'FCFC' (when not a general resource class)
Internal single asterisk X'FBFC80'
Percent sign X'FB'
Ampersand X'FA70'

When the list of members is returned, the format of the output is as follows:

Description Length

Length of member 4 bytes
Member name ? bytes (length determined by length field)

When the list of users is returned, the format of the output is as follows:

Description Length

Length of userid 4 bytes
Userid 8 bytes
Length of access field 4 bytes
Access 1 byte X'80' Alter access
X'40' Control access
X'20' Update access
X'10' Read access
X'01' None

When the conditional access list is returned, the format is the output is as follows:

Description Length

Length 14 bytes
Filler 8 bytes
Length of userid 4 bytes
USRDATA function (list and maintain users' USRDATA fields)

Use the USRDATA function to list, add, update, or remove the USRDATA fields of a user profile.

AUTHORITY

The user must have access to the zSecure CICS Toolkit command. Depending on the function requested, the profile is TOOLKIT.USRL for the list function, TOOLKIT.USRA for the ADD and UPDATE function, and TOOLKIT.USRD for the DELETE function. The user must also have access to the USRN.usrdata-name profile. The affected USERID must be within scope for USRDATA management functions. This means that the user must have access to USRU.dflgrp.

COMMAREA

Minimum size 365 bytes.

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

The API provides three methods of listing the USRDATA of the indicated USER. If you select L for CODE1 and specify a value only for PROF, all USRDATA fields are returned in USRDLS. If you also specify a value for USRN, the value for the specified USRDATA name is returned in USRV. If a value is present in the USRV field, it is used as the first few characters for the requested USRDATA value. For
this last function, zSecure CICS Toolkit supplements USRV with additional characters from the first matching USRDATA Name/Value pair.

Except for the LIST and DELETE functions, zSecure CICS Toolkit does not support duplicate USRDATA names. If you have multiple USRDATA name/value pairs with the same name, you might only inspect and delete them, until the name becomes unique.

When using the LIST function, only those USRDATA names or values are returned for which you are authorized by USRN.usrdataname. If you request all USRDATA names/values, the returned list excludes those items for which you are not authorized.

If you request a LIST of all USRDATA names/values, only those names and values that fit completely within the supplied commarea are provided. In addition, the API_USRD_RC is set to indicate the overflow condition. If you need all values, you must provide a sufficiently large commarea.

**Note:** The space reserved for each USRDATA value is 255 bytes, irrespective of the actual length of the data.

**VERIFY function (verify user ID and password)**

Use the VERIFY function to verify a user ID and password.

**AUTHORITY**

NONE

**COMMAREA**

Minimum size 101 bytes (125 bytes if using Newpass, Termid, or APPL parameters).

In your application, use the CQTMAPIA or CQTMAPIC mapping macros (copybooks) provided in the SCQTMAC library.

```
API_FUNC DC CL4'VERF' Function code for VERIFY.
API_RC DC XL01'00' Return code
API_MSG DC CL79' ' Message area
*  
API_VERF_RC DC XL01'00' Return code from VERIFY.
*  
API_VERF_USERID DC CL08'USERID ' The userid being verified.
*  
API_VERF_PASSWORD DC CL08'PASSWORD' The password for this userid.
*  
API_VERF_NEWPASS DC CL08'NEWPASS' The new password for this userid.
*  
API_VERF_TERMID DC CL08'TERMINAL' A terminal id.
*  
API_VERF_APPL DC CL08'APPLNAME' An application name.
```

The VERIFY does not perform a signon for the user. It only verifies that the password entered for this user ID is correct.

Multiple failures to verify a user ID and password might result in the user ID being revoked, depending on your system parameters.

You might also specify the following optional parameters:
API_VERF_NEWPASS
Specifying a new password changes the users password to the specified new password. The API_VERF_PASSWORD field has to contain the valid current password for the user before the new password works.

API_VERF_TERMID
If a terminal ID is present and TERMINAL checking is turned on in RACF, it verifies the users authority to use this terminal at the current time and date.

API_VERF_APPL
If an application name is present and APPL checking is turned on in RACF, this will verify the users authority to use this application.

This command is only available through the API.

Sample programs
Several example programs are provided for your use as part of the product.

The example described in "Simple API interface" is a general example showing how to use the API interface. The example in "Resource Profile List Interface" is a simple program showing the use of the Resource Profile List API.

Simple API interface
A sample program that demonstrates how the API might be used is provided in the SCQTSAMP data set. This program shows how a resource check might be performed using the RSRX API interface.

The user interface of these programs is simple, and does not perform any validation or additional processing. The examples are intended only to demonstrate the use of the CQTMAPIA area for passing information back and forth between the API and your application program.

To install the sample programs, translate and compile the mapset and the program, and define the resources to CICS. See the following example resource definitions:

```
DEFINE PROG(CQTXAPIR) L(ASSEMBLER) EXECKEY(USER) DA(ANY) GROUP(CQTSAMP)
DEFINE TRANSACTION(XAPI) PROG(CQTXAPIR)
    PROFILE(CQTSAMP) GROUP(CTQTSAMP) TASKDATALOC(ANY)
DEFINE MAPSET(CQTXAMP) GROUP(CQTSAMP)
```

Program CQTXAPIR allows you to enter resource names and resource classes. It then checks to see if you have access to the resource. Alternatively, you might enter a user ID, other than your own, against which to perform the access check.

You can tailor this example to suit your own environment or to do specific editing on any of the fields.

Resource Profile List Interface
An example program that demonstrates how to use the Resource Profile List interface is provided in members CQTXAPIL, CQTXAML, CQTXCPIL, and CQTXCML in SCQTSAMP. The same program is provided in both assembler and COBOL form.
About this task

The programs use a BMS map to display an initial panel, where the user can fill in some filters and options for the RSRL (Resource Profile List) API interface. The source for the BMS map is provided twice to generate different include members, but the two source members are otherwise identical. The programs call the API and show some relevant parts of the output. The programs serve no practical purpose, aside from illustrating how the API can be used, and verifying that the installation has completed successfully.

Procedure

To install the samples programs:
1. Translate, compile and linkedit the mapset and the program.
2. Define the resulting modules to CICS.

Results

See the following example resource definitions:

```
DEFINE PROG(CQTXCPIL) L(COBOL) EXEC KEY(USER) DA(ANY) GROUP(CQTSAMP)
DEFINE TRANSACTION(RSRC) PROG(CQTXCPIL)
    PROFILE(CQTSAMP) GROUP(CQTSAMP) TASKDATALOC(ANY)
DEFINE PROG(CQTXAPIL) L(ASSEMBLER) EXEC KEY(USER) DA(ANY) GROUP(CQTSAMP)
DEFINE TRANSACTION(RSRA) PROG(CQTXAPIL)
    PROFILE(CQTSAMP) GROUP(CQTSAMP) TASKDATALOC(ANY)
DEFINE MAPSET(CQTXAML) GROUP(CQTSAMP)
```

Note:

- To use the example programs, install the group CQTSAMP as shown, and run the transaction RSRA or RSRC.
- To view the output, you might need to use CEBR to view the entire TSQUEUE that is created as part of this program.
- It is the responsibility of the calling program to remove the TSQUEUE after it has been created.
- The example programs do not delete the TSQUEUE after usage, so that you can inspect the data after the transaction has ended.
- After completing your testing, manually delete the TSQUEUE.
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web
sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not
been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM’s application programming interfaces.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

**Trademarks**

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, Acrobat, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium and the Ultrium Logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Other company, product, and service names may be trademarks or service marks of others.
Index

A
accessibility ix
ADDGROUP command 36
ADDUSER command panel 38
ALTGROUP command 36
ALTUSER command 39
CICS segment option 41
OMVS segment option 44 panel 39
TSO segment option 43
WORKATTR segment option 47
API 87
Access Authority Check (Extended) function 91
Access Authority Check function 90
ADDGROUP / ALTGROUP / DELGROUP function 97
ADDUSER function 98
ALTUSER (CICS SEGMENT) function 100
ALTUSER (OMVS SEGMENT) function 102
ALTUSER (TSO SEGMENT) function 101
ALTUSER (WORKATTR SEGMENT) function 102
ALTUSER function 99
CONNECT function 103
DELETE DATASET function 104
DELETE USERID function 105
implementing field or record level security 89
LISTDATASET function 105
LISTGROUP function 107
LISTUSER function 108
PASSWORD function 110
PERMIT function 111
PERMITX function 112
RACLINK function 113
RALTER/RDEFINE/RDELETE function 114
REMOVE function 114
Resource Profile List function 92
RLIST function 115
Sample programs 119
searching the RACF database 89
simple interface 119
USRDATA function 117
VERIFY function 118
API functions
Changing the authorized user 88
using the COMMAREA 87
Application interface overview 1
Application security check application conversion 29
Application security checking alias definitions 31
checking the OP-ID 29
with zSecure CICS Toolkit 29

B
BMS mapsets 22

C
checklist installation 5
post-installation 5
CICS
application security checking for defining programs, mapsets and transactions to 12
Transaction Server 22
updating tables 13
Command interface overview 1
Command interface overview 35
commands
ADDGROUP 36
ADDUSER 38
ALTUSER 39
CICS segment option 41
OMVS SEGMENT option 44
TSO segment option 43
WORKATTR segment option 47
Command interface overview 35
CONNECT 49
DELETE 51
handling API requests 87
LISTDATASET 51
panel 54
toggle option 54
LISTGROUP 57
LISTUSER 63
Main menu 35
PERMIT 72
RACLINK 73
RALTER 75
RDEFINE 75
RDELETE 75
REMOVE 76
RLIST 77
USRDATA 82
CONNECT command 49
Control 85
CQTJACC 9
CQTJALL 8
CQTJAPP 9
CQTJDDD 9
CQTJRDO 12
CQTJRREC 9
CQTJSMPA 8
CQTJSMPB 8
CQTJSMPC 8
CQTJPCNTL
parameter definitions 12
parameter descriptions 23
parameters for zSecure CICS Toolkit 23
parameters verification 26

D
data set profile listing 105
DATASET profiles granting access to 112
removing access to 112
date formatting 2
DELETE command 51
DELGROUP command 36
DELUSER command 98

E
education ix
Exit points
transferring control from zSecure CICS Toolkit 85

F
Field
implementing security for 89

G
Group
adding 97
altering 97
connecting a user or group to 103
deleting 97
listing profile 107
removing a user or group from 114

I
IBM
Software Support ix
Support Assistant ix
IEASVCxx
update 10
information retrieval 32
installation
allocating TARGET and DLIB data sets 8
applying the product 9
checklist 6
creating and initializing SMP/E zones 7
defining options 8
protecting the SVC 10
receiving the product 9
SMP/E 5
updating SMP/E DDDEFs 9
Installation
automatically assigning USS UIDs 19

CSI
defining Global 8
defining Product 8
Installation (continued)
  automatically creating home
directories 19
  defining programs, mapsets and
transactions to CICS 12
  defining SCQTLOAD as
APF-authorized 11
  Enabling/Disabling zSecure CICS
Toolkit 11
  installing the SVC 10
  making the RACF definitions 14
  updating CICS tables 13
  updating the CICS startup JCL 11
  installation checklist 5
  introduction 5
IPL
update 10
IRRPNL00 function 96

N
National Language Support 22

O
OMVS
  assigning UIDs automatically 19
  automatically creating home
directories 19
  online
  publications vi
  terminology vi
  Operator ID check 29
  application conversion 29

P
parameters
  CQTPCNTL values verification 26
Parameters
  descriptions of CQTPCNTL
  parameters for zSecure CICS
  Toolkit 23
  Parameters for 23
  PARMLIB 11
  Password
  changing 110
  verifying 118
  PERMIT command 72
  panel 72
  post-installation checklist 5
  problem-determination ix
  Profile
  delete 104
  listing and maintaining 114
  view details of 115
  publications
  accessing online vi
  list of for this product vi

R
RACF
  defining zSecure CICS Toolkit
  commands to 14
  RACF database
  searching 89
  RACLINK command 73
  panel 73
  RALTER command 75
  RDEFINE command 75
  RDELETE command 75
  reason codes, IRRPNL00 function 96
  Record
  implementing security for 89
  REMOVE command 76
  Resource
  checking user access 90, 91
  granting access to 111, 112
  removing access to 111, 112
  resource access verification 32
  Resource Profile List 120
  Restart 20
  manually 21
  return codes, IRRPNL00 function 96
  RLIST command 77
  Conditional access option 81
  panel 81
  Display option 79
  panel 79
  Members option 79
  panel 79
  Users
  panel 80
  Users option 80
  RRSF 2
  RRSF user id associations
  approving 113
  defining 113
  listing 113
  removing 113
  RSRC / RSRX functions 29
  RTCK transaction
  verifying CQTPCNTL parameters 26
  RTST transaction 21

S
SCQTLOAD 11
  Security checking
  single point 29
  simple
  application security interface 31
  simple application security interface 31
  SMP/E Modification Control
  Statements 9
  SVC
  unauthorized use 10

T
  terminology vi
  toggle option, LISTGROUP command 54
  training ix
  Translating BMS mapsets 22
  troubleshooting ix
  TSQUEUE for profiles 96

U
User
  adding a profile for 98
  authorization for RACF
  commands 14
  changing password 110
  changing the authorized user 88
  changing the CICS segment 100
  changing the OMVS segment 102
  changing the profile for 99
  changing the TSO segment 101
  changing the WORKATTR
  segment 102
  check access to resources 90, 91
  delete id 105
  listing and maintaining
  USRDATA 117
  listing profile 108
  user information retrieval 32
  USRDATA command 82
V
verifying resource access 32

Z
zSecure CICS Toolkit
  manual restart 21
  Security resources 16