IOF

(Interactive Output Facility)

CICS Installation Guide

Release 8G

Copyrights and Trademarks

Triangle Systems, Inc.
P. O. Box 12752
Research Triangle Park, NC 27709
Telephone: (919) 544-0090 Fax: (919) 942-3665
Tech Support Email Address: IOFTech@Triangle-Systems.com
Web Page: http://www.triangle-systems.com

September 2017

Table of Contents

1. Initial Generation of the Product

This chapter describes the process for the initial installation of IOF/CICS. If you have previously installed IOF/CICS see Chapter 5 for information about installing and testing subsequent maintenance releases.

Distribution Libraries are Loaded

This chapter assumes that the Mainframe Product Install (MPI) file has been loaded to your system and executed to create the IOF distribution libraries.

If this is not the case, download IOF from the Triangle Systems web site:

- Connect to http://www.triangle-systems.com
- Click on "IOF Technical Support"
- Click on "Order or Download the Latest Release"

Updating System Libraries

None of the tasks described in this chapter update any of your system libraries. The tasks described in Chapter 2 update your system libraries only with specific copy jobs whose sole purpose is to copy IOF components to system libraries.

IOF Options

You have chosen some simplified options to make it easy to generate an initial testing version of IOF. If you wish to review the entire set of IOF options before continuing:

- Select ISPF option 6
- Exec 'prefix.IOFC8G0.INSTALL(SETIOF)'

Generate the Install Library

Submit the M10INIT job to create the M13GEN job and other IOF INSTALL library jobs:

```
SUBMIT 'prefix.IOFC8G0.INSTALL(M10INIT)'
```

Assemble and Link the Product

Submit the M13GEN job (generated by M10INIT above) to do the required assemblies and link the product load modules into the distribution library.

If you have any problems with this job, please contact IOF Technical Support at:

Triangle Systems, Inc.
P. O. Box 12752
Research Triangle Park, NC 27709
Telephone: (919) 544-0090   Fax: (919) 942-3665
Tech Support Email Address: IOFTech@Triangle-Systems.com
Web Page: http://www.triangle-systems.com

2. Testing and Installing the Initial Generation

This is a description of the steps necessary to install and test the load modules that were generated by the M13GEN job. This chapter is only applicable to the very first time that you install IOF. See Chapter 5 for information about installing and testing subsequent maintenance releases.

Install IOF SVC Module in Link Pack

Copy the IOFSVC load module from the IOF LOAD library to a system LPA or MLPA library.

Update the appropriate IEASVCxx member of SYS1.PARMLIB. For example, if you specified 235 as the SVC number, add the following statement to your IEASVCxx member:

```
SVCPARM 235,REPLACE,TYPE(3),EPNAME(IOFSVC)
```

Please contact IOF Technical Support if you would prefer to use an ESR for authorization.

IPL to Activate the SVC Module

If you copied the SVC module to an MLPA library you must modify the appropriate IEALPAxx member of SYS1.PARMLIB to include the new module name.

IPL your system to activate the SVC module copied above. If you copied the module to an LPA library, you must specify "CLPA" when you IPL. Failure to do so could cause your CICS system to abend when you attempt to invoke the IOF transaction for testing.

Copy Product Load Modules to System Libraries

Submit job M14COPY to copy the product load modules to the system link list library specified in your C64LINK options member. Note that the IOFCvvvM, IOFCvvvA, IOFCvvvU, and IOFCvvvP modules will not execute from a STEPLIB. The M14COPY job will also copy the IOFCIC and IOFCMT modules to the CICS load library specified in your C66CRPL options member.

Refresh LLA after running the M14COPY job.

See Chapter 17 for recommendations about module placement to improve system performance.

Add CICS Resource Definitions

Review the CICS resource definitions in the RDOIOFxx source library member appropriate to your CICS release. Modify these definitions as required and add them to your CICS system using your normal CICS resource definition procedures.

Reviewing CICS System Initialization (SIT) Parms

IOF/CICS operates as a conversational task under CICS. If you are using the MXT parameter to control the number of CICS tasks, you should increase your MXT value by the maximum number of concurrent IOF users you expect to have.

If you do not place modules IOFCvvvM, IOFCvvvA, IOFCvvvU, and IOFCvvvP in the LPA as explained in in Chapter 17, the modules will be loaded by the operating system into the CICS address space but outside the CICS dynamic storage areas (DSA/EDSA). Only the relatively small IOFCvvvA module will be loaded below the 16MB line. You should review your CICS region size, and the CICS SIT parameters related to OSCOR, DSA, and EDSA sizes to insure there will be sufficient memory left in the address space to load any IOF modules which you choose not to place in the LPA. You can determine module sizes by examining the linkage editor output from the M14COPY job.

Even if you do install these modules in LPA, you may still wish to adjust the above named parameters slightly to leave some unused memory in the CICS address space. IOF/CICS does not perform MVS GETMAINs, but as with many CICS transactions, small amounts of memory outside the DSA may be obtained on its behalf for operating system control blocks, etc.

Set Up to use the TSITRACE Command

We recommend that you add the following DD statement to your CICS startup JCL:

```
//$IOFLOG$ DD  SYSOUT=c
```

where "c" is any valid sysout class or "*". Output from the **TSITRACE** command will be written to this DDname. See Chapter 15, IOF Trace Facilities, for detailed information about the **TSITRACE** command.

If you do not allocate this data set in your CICS startup JCL, you will have to use the sample ADYN transaction supplied with CICS or an equivalent program to dynamically allocate it before you can use **TSITRACE**.

Set Up Help Library

If you wish to pre-allocate the IOF help data set, you can add the following DD statement to your CICS startup JCL:

```
//ddname  DD  DISP=SHR,DSN=prefix.IOFCvv0.HELP
```

where "ddname" is the value from options member B86HLPDD, "prefix" is the value from options member C50PREFX, and "vv" is the IOF version (8G).

If you omit the DD statement, IOF will use the value from options member A86HLPFX to form the name of the help data set and then dynamically allocate it. This will simplify the testing of user modifications and future IOF releases in parallel with your production release since IOF will automatically locate the correct **HELP** library.

Set Up Profile Library

Add the following DD statement to your CICS startup JCL:

```
//ddname  DD  DISP=SHR,DSN=prefix.IOFCv00.PROFILE
```

where "ddname" is the value you specified in options member B74PRFDD (normally IOFvPROF), "prefix" is the value you specified in options member C50PREFX, and "v" is the first digit of the IOF version number.

Understanding Profiles

When you start the IOF transaction, IOF will attempt to read the profile member associated with your userid. If the profile member is found, IOF will proceed to display the *IOF Option Menu* or the *IOF Job List Menu*. If the member is not found and you coded "PROFREQ YES" in options member B76PRFRQ, the IOF session will be terminated with an error message. If the member is not found and you coded "PROFREQ NO", the session will continue and will use your installation's default profile values. If you override any of these installation defaults during the session, a new profile member automatically will be created and these new values will be remembered for your next session.

To facilitate initial testing, a special "bootstrap" profile member is supplied in the profile data set which was loaded from the MPI file. This profile member is named $MASTER and has both "ACCOUNT" authority and "OPERATOR" authority. ACCOUNT authority will permit the definition of profile members for other users. See Chapter 20 for more information. OPERATOR authority will permit access to any job in the system and the use of other powerful features not available to the ordinary IOF user. See Chapter 9 for more information.

This special profile member is provided only as an aid to installing and testing the product and as a means of initially giving ACCOUNT authority to the users in your installation who will be responsible for maintaining IOF profiles.

For security reasons you should use the profile management facilities of IOF to delete the $MASTER member once you have given ACCOUNT authority to at least one user and have verified that this user can access the IOF profile management functions by selecting option **PM** on the *IOF Options Menu*.

To cause IOF to use this "bootstrap" profile, you must specify the special operand **/$MASTER** when you invoke the IOF transaction.

You can add your own userid to the profile data set and give yourself ACCOUNT and OPERATOR authority by entering the following sequence:

```
IOF /$MASTER      Start IOF using the special
                  "bootstrap" profile and display the
                  IOF Options Menu
PM                Invoke the profile manager panel
MODEL $MASTER     Read the $MASTER member and use it
                  as a model
ADD   userid      Add your userid to the profile data
                  set
END               Return to IOF Options Menu using
                  PF3/PF15
END               Return to CICS using PF3/PF15
```

Perform Initial Testing

Review your A70FLOW option member and note the value you specified for the "INITCMD" option. If you specified "INITCMD=INITMENU," enter **IOF** from a clear screen to display the *IOF Option Menu*. If you specified "INITCMD=INITCMD," enter **IOF *** to display the *IOF Option Menu*. From this menu you can select displays for jobs, output groups, or devices.

As a simple test, position to the JOBNAME field of the *IOF Option Menu* and enter a generic job name that you know will match more than one job in your system (for example, entering **PROD*** will display all jobs beginning with "PROD").

You should see a list of all jobs matching the generic job name. Place an **S** beside one of the jobs and press **ENTER**. You should see a display of the return codes for the job followed by a menu of the sysout data sets for the job. Place an **S** beside one of the data sets, and you should enter browse for that data set.

Enter **END** (PF3) to return to the sysout data set menu. Enter **END** again to return to the list of jobs. **END** again will take you back to the *IOF Option Menu*. From there you can select another option or enter **END** again to return to CICS.

You easily can select a new IOF option from anywhere within IOF by preceding the desired option with a "**/**". For example, you can enter **/M** anywhere in IOF to get immediately to the *IOF System Monitor Display*. The ***IOF Quick Reference Summary*** card provides information to assist you in

further testing of IOF features designed for systems operators and support personnel. Chapter 2 of the ***IOF User Guide*** contains a detailed sample session.

To see the jobs that belong to you, press **ENTER** on the *IOF Option Menu* without entering any other parms. See options member A40SCOPE for a description of the jobs that are considered by IOF to belong to you. Depending upon your A70FLOW option, you can also go directly to your jobs from outside IOF by entering **IOF** or **IOF *** from a clear CICS screen. See Chapter 9 for more discussion of job ownership options for IOF.

You can reverse the meaning of **IOF** and **IOF *** using option A70FLOW. You also can reverse their meanings for selected user groups using the INITCMD= parm on GROUP macros in options member B23ALLOW. This might be desirable for many end users since they may wish to look at only their own jobs. IOF is structured with all jobs on a single menu, so end users normally will not need to use the *IOF Option Menu*.

Activating IOF Sysplex Features

If you have IOF/TSO and are running a sysplex, See Chapter 27 for a description of IOF features that can help you manage your sysplex environment. Some of these features have additional installation requirements.

Changing Execution Options

If you discover during testing that you want to review or change an IOF option, you can edit that member in the OPTIONS library.

You can also run SETIOF from ISPF option 6 by entering:

```
exec 'prefix.IOFC8G0.INSTALL(SETIOF)'
```

To install changed options, submit the M13GEN job from the INSTALL library followed by the M32COPY job.

Contact Us

Please contact IOF Technical Support if you have any questions during the testing process. We have a number of optional ways to configure IOF to satisfy the needs of your users. There are also many ways that an individual user can tailor IOF to their own personal tastes.

3. Installing Options Changes

Follow the steps below to incorporate options changes into the IOF load modules.

Generate the New Load Modules

Submit job M13GEN from the IOF Install library to perform the necessary assemblies and produce the product load modules.

Install Test Versions of Load Modules

Submit job M56TEST to copy the new load modules to the system library (specified in C64LINK) and the CICS DFHRPL library (specified in C66CRPL) with new names that will not conflict with your production module names. Note that these modules will not run from a STEPLIB. To test the new modules, use transaction code **IOF#** instead of **IOF**.

Install New Modules as Production

After testing is completed, submit job M14COPY to copy the new load modules to your system and CICS libraries as the production versions. If you want to preserve the previous production versions of the load modules, rename them before. You must refresh LLA after running the M14COPY job.

4. Options Changes Requiring Abbreviated Generation

All options changes now require the installation process that is described in Chapter 3.

5. Installing a New Release of IOF

This chapter describes how to install a new release of IOF after the product has previously been installed. The new release can be installed and tested in parallel with your production version, and can be made the production version after testing is complete.

To install the new release from our web site, see the instructions at:

- http://www.triangle-systems.com
- Click on "IOF Technical Support"
- Click on "Order or Download the Latest Release"

6. Testing a New Version in Parallel with Production

This is a description of the steps necessary to install a new version of the product for testing in parallel with your existing production version. This procedure assumes that the product load modules have already been generated by the M13GEN job.

Test Environment for Product Load Modules

Submit the M32COPY job to copy the product load modules that must reside in a system link list library into the library specified in your C64LINK options member. These module names will not conflict with the current production module names since these load module names contain the version and level id for this new release of the product.

The IOFCIC load module is not copied by the M32COPY job above. If you copied it into your CICS load library, it would become your current production version of this module. Since the IOFCIC load module is internally programmed to invoke a specific version and level of IOF, any user who requested the IOF transaction would invoke the new IOFCIC. This would, in turn, invoke the new load modules copied above with the M32COPY job. In effect, you would have installed the new version as your production IOF.

To avoid this conflict, testing must be done using an alternate transaction and program name. Submit the M57NEW job to copy the new IOFCIC load module to the CICS load library specified in your C66CRPL options member. The module will be copied as IOFCICN and will be invoked when you enter the "IOFN" transaction.

It is important to point out that this test procedure will not work if the M32COPY job above is not executed. The load modules copied by M32COPY will not execute from a STEPLIB and must be run from a system link list or LPA library.

Testing the New Version

To test the new version of the product, use the "IOFN" transaction rather than "IOF". IOFN will invoke IOFCICN which, in turn, will invoke the new version of the product.

Changing "A" or "B" Options During Testing

If you discover during testing that you need to change one or more "A" or "B" options, you can modify the associated options members and regenerate the load modules. The type of generation needed depends on the options that you changed.

Each "A" and "B" option is labeled as requiring either a "full generation" or an "abbreviated generation". If you have changed any option designated as requiring a full generation, run the M13GEN job to rebuild the load modules. Then, run M32COPY and M57NEW to copy the load modules into your system and CICS libraries. See Chapter 3 for more information about "full generation" options.

If all the options that you changed are "abbreviated generation" options, run job M18NEWOP to perform the abbreviated generation and copy the options load module into your system library. Comments in the options members clearly indicate whether they are "full generation" or "abbreviated generation" options. See Chapter 4 for more information about "abbreviated generation" options.

Changing "C", "D", or "K" Options During Testing

If you discover during testing that you need to change one or more "C", "D", or "K" options, you should modify the associated options members, run the M10INIT job to recreate the install library, and run M13GEN to rebuild the load modules. Then, run M32COPY and M57NEW to copy the new modules into your system and CICS libraries.

Installing the New Load Modules as Production

Submit the M33COPY job to copy the CICS transaction modules into your CICS load library. At this time they will replace your previous production versions of those modules and the new modules will become the production versions.

If you want to preserve the old load modules, rename them in your CICS load library before submitting the M33COPY job.

7. JES2 Maintenance Considerations

When You Apply JES2 Maintenance

In most cases JES2 maintenance does not affect the operation of IOF. If you do notice problems after applying JES2 maintenance, run the M13GEN job to pick up the maintenance. Then, run the M32COPY job to copy the updated load modules to your system library.

When You Install a New Version of JES2

If you are installing a new version of JES2 (with a new FMID), you will need to re-install the IOF load modules. First, call IOF Technical Support or check the IOF Technical Support section of our website to determine if a new product version is available that is at a higher level than your current production version. You can download MPI files or maintenance required for new JES2 versions directly from the web site. The IOF Technical Support web address is: http://www.triangle-systems.com.

If a Higher Level Release is Available

If a higher level IOF release is available, order or download the new version and follow the instructions in Chapter 5.

After loading the new distribution libraries, update options member C75ASMJS to reference the source libraries for your new version of JES2 before running the M10INIT job. See Chapter 5 for a more information about this procedure. If you are creating a new target MVS system, you also will need to update options member C64LINK before running M10INIT.

If You are Currently Running the Latest Level

If you are currently running the latest maintenance level of the product, follow the instructions in Chapter 13 to create a new set of libraries that can be used as a base for generating the new IOF load modules.

After completing the procedures described in Chapter 13, submit job M13GEN in the new install library to do the necessary assemblies and generate the product load modules.

If you are creating a new target MVS system, you can copy the product load modules to the target link list library and the new CICS load library by submitting job M14COPY. When you start CICS on the new target MVS system, you will be using the new version of IOF.

If you are installing a new JES2 on your current production MVS system, you should refer to Chapter 6 for information about testing the new version of IOF in parallel with your production version.

8. When You Install a New Version of CICS

Testing Your Old IOF With Your New CICS

You normally will not need to re-generate the IOF load modules when you install a new version of CICS. Add a DD statement for the profile data set and, optionally, the TSITRACE data set which was discussed in Chapter 2. Make sure that IOFCIC and IOFCMT load modules can be found in the DFHRPL concatenation of your new CICS system (see option C66CRPL). Then, test IOF with the new CICS. If you have any problems or questions, contact IOF Technical Support.

If your testing is successful, update option members C76ASMCI and C77LNKCI to reference your new CICS libraries; run M10INIT to re-generate the IOF install library. This will change all jobs in the install library that have references to your CICS libraries so they will be correct the next time you need to make options changes or install a new version of IOF.

Re-generating IOF for Your New CICS

If you wish to change IOF options at the same time you install your new CICS or simply would feel more comfortable working with a new set of libraries, you must first follow the instructions in Chapter 13 to create a new set of libraries that can be used as a base for re-generating the IOF load modules.

Before running the M10INIT job in the new install library, the following changes must be made in the new options library. Change option member C66CRPL to point to a different load library so that you inadvertently do not replace your production IOF. Also, update option members C76ASMCI and C77LNKCI to reference your new CICS libraries.

After making the necessary options changes in the new options library, run the M10INIT job from the new install library to create new install jobs. Run the M13GEN and M14COPY jobs from the new install library and check their output. Make sure that the JCL for the new CICS system has a DD statement for the profile data set and that the new load library specified in option member C66CRPL is in the DFHRPL DD concatenation.

9. Access Control Overview

By default IOF allows most users to control only their own jobs, while a user with OPERATOR authority is allowed to control all jobs in the system. You have the ability to change the default rules to match the requirements of your installation.

This section describes some simple procedures that can be used to maintain the access rules for IOF. See Chapter 26 for more detailed information about IOF access control.

SDSF Considerations

This topic should be skipped unless you previously used IBM's SDSF product to view output under TSO. The initial install process, described in Chapter 1, includes the ability to convert your current SDSF ISFPARMS data set to comparable IOF access control rules. If you chose that option, a new B23ALLOW options member was created for you that contains the GROUP, ALLOW, and LIMIT macros necessary to simulate your current SDSF access control environment.

You may wish to review the new B23ALLOW options member:

- One IOF GROUP macro is generated for each ISFGRP macro. Permissions are granted by ALLOW macros that point back to the GROUP macros.
- The default IOF options menu for end users is OPTUS1. This menu is used instead of the system programmer default (OPTOPT) if the SDSF group does not specify either PR, DA, INIT, or LOG in the AUTH= parameter. You can change this by changing the PANEL= parm on the GROUP macro.
- All users are allowed to control the jobs they submitted. To change this, modify the ALLOW macros labeled MY1, MY2, MY3, MY4, and MY5.

If you are using the IBM defined SAF classes JESSPOOL, OPERCMDS, WRITER and SDSF to control access to SDSF resources, IOF can honor your existing SAF rules. Access to IOF functions, commands and displays will be virtually identical to the SDSF access to the same features. To cause IOF to honor the IBM-defined SAF rules, specify "IBMSAF=YES" in the A60ACF option. Note that IOF LIMIT macros will always be strictly enforced, even if IBM SAF rules are being used.

Important note: If you enable the JESSPOOL, OPERCMDS, WRITER and/or SDSF SAF classes without proper rules and profiles in place, you may inadvertently permit access to IOF resources.

Adding New Access Control Rules

If you have only the CICS version of IOF, you must add new access control rules by directly editing the ALLOW and LIMIT macros in options member B23ALLOW. However, IOF/TSO provides a simple ISPF dialogue interface to assist you in building new ALLOW and LIMIT macros. If you have both versions of IOF, you can copy your IOF/TSO access control options into IOF/CICS.

Enter the **ALLOW** command from any IOF panel under ISPF to invoke the dialogue. You will be prompted by a series of ISPF panels for the information necessary to build ALLOW and LIMIT macros. You optionally then can have the new ALLOW/LIMIT macros appended to your production B23ALLOW options member.

Even if you do not use this mechanism to update your B23ALLOW member, it is still a very good way to learn how ALLOW and LIMIT macros are used.

Deleting Access Control Rules

Edit options member B23ALLOW to delete access control rules. You will normally be deleting or modifying an ALLOW or LIMIT macro.

Using RACF, Top Secret, or ACF2 to Control IOF Access

IOF allows you to control access with your external security system:

- Use options member A60ACF to specify which security system you have and whether operators and started tasks should be given access without requiring rules in the security system.
- Use options member B24ACFDF to select the types of access you want to control with your security system.
- IOF/CICS users should review Chapter 25 for important information about IOF security in a CICS environment.

IOF checks the IBM defined JESJOBS class before canceling jobs. **If you enable the JESJOBS class, you should also add rules or profiles to control access to cancel jobs**.

Access Control Trace

To see exactly how IOF validates access to IOF functions, use the IOF TRACE facility described in Chapter 15.

10. Attributes and Options for User Groups

Each IOF user is assigned to a group at the start of each IOF session. Options member B23ALLOW defines which users belong to each IOF group. In addition to indicating which jobs, output groups, devices, etc. that the users are allowed to access, group membership also has certain other implications.

You can specify the following attributes for groups of users using GROUP macros in the B23ALLOW options member:

- What jobs are displayed on their default *IOF Job List Menu*.
- Whether the user's TSO session is to be included on their default *Job List Menu*.
- A limit to the number of sysout records that can be scanned in a single **FIND** command.
- A minimum time delay between times that the user can hit **ENTER** to refresh their *IOF Job List Menu*.
- Whether the user is allowed to use the **EXTEND** command for the *IOF Job List Menu*.
- A minimum pause interval for the **EVERY** and **PAUSE** commands. Or, you can disable these commands.
- The WTOR route codes for action messages to be displayed at the bottom of the system log browse.
- Default options for the *IOF Monitor Display*, including the ability to prevent access to the display.
- The default system id for system log browse.
- Alternate display formats, described in Chapter 11, for certain display panels.
- Whether the user is to enter the *IOF Job List Menu* or the *IOF Option Menu* when they invoke IOF with no parms.
- An alternate options menu to display options and accept session parms.
- A special subset of the global commands table that applies only to users in the group.
- Whether the user is allowed to use the DR command.
- Whether the user is allowed to use the INPUT command on the IOF Job Summary.
- The specific options that will be displayed on the *IOF Option Menu* for members of the group.

11. Controlling Display Formats

Users can tailor most IOF display formats to fit their personal needs and preferences using the **CUT**, **PASTE** and **ARRANGE** commands. Panel modifications are saved in the user's profile until deleted. Enter "HELP ARRANGE" for a description of **CUT**, **PASTE** and **ARRANGE**, or see *Chapter 6, Customizing IOF Panels*, of the **IOF User's Guide**.

If you need to globally change the default formats on one or more IOF panels, options member B68GORMT tells you how. By changing B68GORMT you can alter the default display formats for all users. By creating new SECTION macros and pointing to them with the FORMATS= operand of GROUP macros, you can select different display formats for different groups of users.

You can also change the default sort order for specific sections with SECTSORT macros.

See options member B68GORMT for a description of the SECTION macro and its relationship to the GROUP macro. Modifying the B68GORMT options member requires an abbreviated generation as detailed in Chapter 4 of this guide.

12. Installing and Maintaining Source Mods

Before deciding to make a change to one of the product source modules, you should carefully investigate the possibility that your requirement can be met by using some combination of product options. Each release of IOF contains new options that can be used to eliminate user modifications. Please contact IOF Technical Support if you are considering source changes. We will be happy to help you find a way to accomplish your objectives without source modifications.

This is a description of a procedure for modifying the product that makes it possible to carry forward your source modifications to new releases. This procedure is not necessary if you are only changing options in the options library. In that case, you should refer to Chapter 3 or Chapter 4 for instructions.

Before making any source modifications to the product, you should create a new set of libraries. This will enable you to create, test and maintain your modifications independently from the base libraries. See Chapter 13 for the procedure to generate a new set of libraries.

Do not proceed to the steps below until you have followed the instructions in Chapter 13 to generate a new set of libraries. You must follow this procedure in order to receive technical assistance with your modifications. If you choose to skip this step and update the source directly, we cannot assume the responsibility for helping you carry forward your changes to the next release level of IOF. We will not be able to help you because it will be impossible to identify and extract the source changes from the old release in order to apply the same changes to the new release.

Please contact IOF Technical Support if you have any questions about this policy. We will try to help you understand the problems created when modifications are developed that cannot be easily identified and applied to subsequent releases. We will be glad to answer any technical questions about the update procedure itself.

Create Source Updates

Each source update should be created as an IEBUPDTE input data set and stored in the newly created version of the product updates library. The member names in the updates library will be referenced in control statements described below.

Each member of the updates library should only contain source updates for a single source module, and each update member should include exactly one "./ ADD" or "./ CHANGE" statement. However, multiple update members may contain updates to the same source module.

Add a Control Statement for each Update Member

Edit the options library member D55UPSRC to add a %VUPDATE or %VADD statement for each update member that you have added to the updated library. The comments in member D55UPSRC describe the %VUPDATE and %VADD control statements.

Create the Job to Update the Source Library

Submit the M52UPSR# job to generate job M52UPSRC which can be used to apply all of the source updates that were described in options member D55UPSRC. If you need to add other members to the updates library, you should update options member D55UPSRC and rerun M52UPSR# to recreate the M52UPSRC job.

It is not necessary to rerun the M52UPSR# job if you only are changing an existing member of the updates library. You need to rerun M52UPSR# if you have added, deleted, or renamed members in the updates library. Remember that options member D55UPSRC must accurately reflect the contents of the updates library or job M52UPSR# will not generate the correct update steps.

Update the Source Library

Submit job M52UPSRC to apply your updates to the source library. The updates will begin with the source members from the original distribution library and update them into your current source library.

Thus, you can rerun the M52UPSRC job at any time and it will go back to your original distribution source, apply all of your updates, and store the updated modules into your current source library.

Assemble Affected Source Modules

If only the JESCTL or OPTIONS source members are affected, you can submit the M13GEN job to assemble the modified source modules and produce the product load modules. In that case proceed directly to the section below, ***Test the New Product Load Modules***.

If you have made source changes that affect source modules other than JESCTL and OPTIONS, you should submit job M70ASM# to generate an assembly job for each source module in the source library. Refer to install

library member M00INDEX for a list of the names of assembly jobs for the various source members. The member names start with M71JESCA.

Submit the generated assembly jobs for the source members (other than JESCTL and OPTIONS) that you need to assemble. Then, submit job M13GEN to assemble JESCTL, OPTIONS, and generate the product load modules.

Test the New Product Load Modules

See Chapter 6 for information about testing a new version of the product in parallel with your production version.

13. Generating an Installation Version of the Product

This is a description of how to create a new set of product libraries that can be used to apply installation modifications to the product.

Each set of product libraries has a three character version and level identification. The first two characters represent the base. The third character is the level identifier and is always "0" for the libraries that are initially loaded the MPI file.

For example for Release 8G, the base version identifier would be "8G" and the last character would be "0". So, the complete version and level identifier would be "8G0".

The third character, or level identifier, can be used by the installation to indicate local levels of the product that are based on a particular release. Using the above example, an installation could create a set of libraries with a level identifier of "1". In this case the complete version and level identifier would be "8G1".

If you already have the product installed in production with a certain set of changes applied (say at level "8G1") and you want to make some additional changes, you can create another installation level (say "8G2") to begin your new changes.

This mechanism allows you to proceed with modifications in an orderly fashion without impacting your current production product or the libraries from which it was generated.

The steps below describe how to create a new installation level of the product libraries.

Select an Installation Level Identifier

Select the one-character installation level identifier. If this is the first installation level since this version of IOF was installed, it would normally be designated as "1." You can use any alphanumeric character that you wish as long as you have not already used it for this version of IOF.

Create the Job to Generate the New Libraries

Edit job M50DISK# to specify your selected installation level identifier in the LEVEL operand of the %VGENJOB statement (the last record in the job). Save the new M50DISK# job; then, submit it to create the M50DISK job.

Carefully Review the Generated M50DISK Job

Review the M50DISK job just created to make sure the data set names of the new libraries to be created match the level identifier you selected. Be very careful here. If you run the M50DISK job with data set names that actually match an existing version of product libraries, it will delete all of the existing libraries.

The M50DISK job first deletes the new library names to make it easy for you to rerun the job if for some reason it fails after partial completion. But, this means it can delete a complete set of existing libraries if you make a mistake here.

Create the New Libraries

Submit the M50DISK job to allocate a new set of libraries with the new installation version identifier and copy the current libraries to the new libraries. When this job completes successfully, you will have a new set of product libraries that match the new installation version identifier.

New Versions of JES2

If the new version of IOF is being generated for a new version of JES2, update options member C75ASMJS in the new options library to reference the macro library for your new version of JES2 (SYS1.SHASMAC).

New Target System

If the new version of IOF is being generated for a new MVS target system, update options member C64LINK in the new options library to point to the linklist library for the new target system.

Generate the New Install Library

Submit job M10INIT from the new install library to build the installation jobs in that library. These jobs will be set up so that you can generate and install the product from the new set of product libraries.

Since all of your previous options will be preserved, you will not need to modify the new options library unless you wish to change an option you had specified previously.

Your new set of product libraries is now allocated and initialized. These libraries can be used to generate a new version of the product without disturbing any previous versions of the product libraries.

Note that a small change has been made to M50DISK beginning with release 7F. Now, M50DISK always copies all IOF data sets from the current level to the new level. In older IOF releases, the SOURCE and OBJ data sets were always copied from level 0. The effect of this change is that IOF maintenance and user mods applied at any level will always be carried forward to new levels.

14. Holding Job Printout for IOF

To review the results of a job with IOF, its output must be prevented from printing before the user has a chance to review it. There are two basic approaches that can be used to accomplish this.

The first approach is to simply hold the sysout data sets for a job. In this case, after reviewing the job with IOF, the user can cancel or release it for print.

As an alternate approach, your installation can add a dummy symbolic destination name (such as "TSO", "FETCH", etc.) to the JES2 initialization parms. This symbolic name can be associated with an unused remote number (Rnn) or with an unused local device (Unn). Users can then route their jobs to this destination (with /*ROUTE statements) to prevent them from being printed.

After reviewing such a job with IOF, users can request that IOF route the job's output to a real JES2 print destination (LOCAL, etc.) with the IOF "PRINT" function. IOF provides a profile option that allows users to supply their default real print destination, eliminating the need to supply the destination each time that they ask IOF to print a job. To set a default print destination, the user enters "P.1" on the *IOF Option Menu*.

15. IOF Diagnostic Aids

IOF Technical Support is available to assist you if you have trouble installing, tailoring, or running IOF. Additional help can be obtained from the IOF virtual help desk on the Internet at http://www.triangle-systems.com. Many common problems can be solved quickly and easily without assistance using the virtual help desk.

IOF Abends

You may have to contact IOF Technical Support if you experience an abend situation. IOF normally produces a seven to twelve line diagnostic area whenever it abends. This diagnostic information displays at the terminal and also in SYSLOG. The diagnostic information displayed is often all that is required by IOF Technical Support to diagnose a problem. If you are making modifications to IOF exits or source code, knowledge of the diagnostic area format will also be useful to you.

```
     IOF 8G0 ABEND DIAGNOSTIC AREA                          99200.1423
1 0000   840C1000  001107E2  001D050C  000CFFFF  *d......S........* <--- abend, coded PSW/R14
2 0010   05985DDC  D01407FE  000090EC  D00C0700  *.q) :........:...* <--- PSW addr, 12 PSW bytes
3 0020   071C0000  85985DE2  00020001  00054F00  *....eq)S......|.* <--- PSW, lng, intcode, xadr
4 0030   00000000  05A6FE66  05A6FE28  05A6FE28  *.....w...w...w..* <--- R0  R3  Regs at time
5 0040   00000001  00000000  05A98250  05A982BC  *.........zb&.zb.* <--- R4  R7  of abend.
6 0050   0598586C  8598DEFC  8598DDCE  05A6EE80  *.q.%eq..eq...w..* <--- R8  R11
7 0060   0598DD08  05A6F50C  8598D214  05985DE0  *.q...w5.eqB..q):* <--- R12 R15
```

Line 1 contains the abend code, the coded PSW and coded R14 at the time of the abend. The PSW and R14 are coded so that they can be used to determine where the abend occurred in IOF without the need of a link edit map. In the first example above, line 1 col 1 contains 840C1000 indicating an S0C1 abend occurred. The coded PSW is 001107E2 which indicates that the PSW was in the csect GLOBCMND at displacement 7E2 at the time of the abend. The coded R14 is 001D050C which indicates that the contents of R14 is an address that points into the csect EASYINP at displacement 50C.

Line 2 contains the twelve bytes of data surrounding the PSW. Lines 3 through 7 contain the PSW, instruction length, interrupt code, translation exception address, and the 16 registers in effect at the time of the abend. If the abend occurs inside an SVC, there will be 5 additional lines containing another set of PSW and register that look like lines 3 through 7.

The following list provides some of the more common csect codes for user modifiable code:

```
JOBACESS  0704xxxx  JESCTL  0411xxxx  SELSETUP  0420xxxx
```

where xxxx is the displacement inside the csect.

Determining the IOF Version

The **VERSION** command can be entered from any IOF panel to display the version of IOF that is currently being executed in the short error message area at the top of the screen. The **HELP** command will then display a long message which includes the date and time the user options module was link edited.

IOF Trace Facility

IOF has extensive trace facilities to help you determine why specific users are allowed (or not allowed) to perform specific IOF functions. IOF group assignment, function validation, and links to the system security system are examples of important functions that may need to be traced in a specific situation.

IOF trace information is written to an output file with a DD name of $IOFLOG$. This file must be allocated before the trace can be started. The **TSITRACE** command is used to activate tracing and select tracing options.

TSITRACE Command

The **TSITRACE** command is used to start or stop tracing and allows you to select specific types of trace entries:

```
TSITRACE[GROUP/INIT/VALIDATE/ALLOW/ACF]
[/OFF]
```

Specifying one or more of the GROUP, INIT, VALIDATE, ALLOW, and ACF parms activates IOF tracing and selects the type of tracing desired:

GROUP	Traces the assignment of a user to an IOF group. See the below examples.
INIT	Traces global permissions assigned based on session start parms. (JOBNAME, DEVICE, etc.)
VALIDATE	Traces requests to permit specific IOF functions.
ALLOW	Traces evaluation of individual ALLOW macros.
ACF	Traces all calls to external security system (RACF, etc.)

Specifying "OFF" turns off tracing and closes the trace data set.

All examples below assume that a trace data set has been allocated to DD name $IOFLOG$.

Example 1. Trace the assignment of a user to an IOF group.

From any IOF panel enter:

```
TSITRACE GROUP
IOFNEST
END
TSITRACE OFF
```

Example 2. Determine why a user can (or cannot) specify a particular job name.

From the *IOF Option Menu* enter:

```
TSITRACE INIT ALLOW
```

Key the desired job name in the JOBNAME field and press enter.

```
TSITRACE OFF
```

Example 3. Determine why a user can (or cannot) select a particular job for review.

From the *IOF Job List Menu* enter:

```
TSITRACE  VALIDATE ALLOW ACF
```

Select the job in question

```
TSITRACE OFF
```

Displaying IOF Options and Variables

IOF has many displayable options and variables. The **DVAR** command lists many of the options that have been selected, and documents the IOF option library member name in which the option is set. It will be instructive to enter **DVAR** to see the options that have been selected for a user.

Syntax

DVAR /iof-var-name/

---

iof-var-name. Any IOF variable name, including variable names that are defined by the user with **SETPVAR** and **SETLVAR** commands. If no parm is specified, several menus of IOF variables will be displayed.

DVAR displays several columns of information for each variable:

NAME The IOF variable name

VALUE	The current value of the variable
FROM	The pool from which the variable is fetched
OPTION	The IOF OPTION member in which the variable is set
Description	The description of the variable

Determining Where IOF Modules Reside

The IOFCIC module must reside in a CICS load library that is part of the CICS DFHRPL concatenation. All other IOF modules must reside in a LINKLIB or LPALIB. IOF modules can sometimes exist in multiple libraries which can be confusing when applying maintenance or updating the product.

IOF/CICS users must invoke IOF with the FINDMOD parameter. From CICS enter:

```
IOF /FINDMOD       or
IOF /FINDMOD(M)    to locate the "main" module
IOF /FINDMOD(A)    to locate the "auxiliary" module.
IOF /FINDMOD(U)    to locate the "user options" module.
IOF /FINDMOD(P)    to locate the "panel" module.
```

16. Dumping the JES2 Control Blocks

The **DUMPCB** line command can be used on the *IOF Job List Menu*, *Output Group Display,* or *IOF Job Summary* to dump the control blocks for a job, output group or data set. Enter the **DUMPCB** primary command on the *IOF Job Summary* to dump the control blocks for that job. You will be placed in IOF browse with the JES2 control blocks displayed. You can use any of the normal IOF browse features to scroll, find character strings, or SNAP information to a printer or external data set.

The following operands are supported on the **DUMPCB** command:

- **DATA(JES2-data-set-number)**. Specifies the internal JES2 sysout (or sysin) data set number for a spool data set whose data blocks are to be displayed in dump format.
- **MTTR(JES2-mttr)**. Specifies the JES2 "MTTR" of a spool block to be dumped. The full eight character JES2 MTTR must be specified.
- **WIDE**. Specifies that wide dump format should be used if IOF is running on a narrow (80 column) terminal. This option is useful if you want to SNAP the data to a wide printer while using a narrow terminal.

For example, to dump the data block at disk MTTR address 01023F07:

```
3 DUMPCB MTTR(01023F07)
```

The **DUMPCB** line command on the *IOF Job Summary* displays the appropriate control blocks for the selected sysout data set. DUMPCB cannot be used from the Job Summary in a CICS environment.

17. Performance Considerations

The suggestions below are optional but are recommended for running IOF in a production environment.

The module IOFCvvvM (where vvv is the version identifier) should be placed in the pageable link pack area.

The module IOFCvvvU should not be placed in the link pack area since it contains most of the product options (including the expiration date).

The remaining load modules (IOFCvvvA and IOFCvvvP) optionally can be placed in the link pack area for improved performance. The IOFCvvvA module operates in 24-bit addressing mode, so it cannot be placed in the extended link pack area.

Although the IOFCIC and IOFCMT modules (the only modules loaded by the CICS loader) are reentrant, they are very small. We do not recommend specifying to CICS that they are eligible to reside in LPA.

18. Product Load Module Naming Conventions

There are two categories of load modules used in IOF/CICS. In the first category are modules whose names must be known to CICS and whose associated transaction codes must be known to end users. These transaction/program pairs are:

```
IOF   -  IOFCIC    Normal production IOF transaction.
IOF#  -  IOFCIC#   Alternate transaction for testing
                   option changes.
IOFN  -  IOFCICN   Alternate transaction for testing
                   user modifications and new
                   releases.
IOMT  -  IOFCMT    Optional IOF Master Terminal
                   Command.
```

These transactions and programs must be defined to CICS using the resource definition facilities of CICS. The load modules must reside in a CICS load library (i.e. a library concatenated with the DFHRPL DD-statement in your CICS startup JCL).

In the second category of modules are the IOFCvvvM, IOFCvvvA, IOFCvvvU, and IOFCvvvP load modules that must be invoked from a link list library (or the link pack area). The "vvv" in the module names represents the version identifier for the modules. End users and the CICS tables do not need to know the names of these modules as their names change with each new version of the product.

When a particular version of the product is generated, the IOFCIC load module for that version is initialized to invoke the correct IOFCvvvM, IOFCvvvA, IOFCvvvU, and IOFCvvvP modules for that version. This means that even though the IOFCIC load module name will always be the same, it is always internally associated with a specific version identifier of the product.

For example, the IOFCIC load module that is generated for version "8G0" will always invoke modules IOFC8G0M, IOFC8G0A, IOFC8G0U, and IOFC8G0P.

This naming convention allows a simple testing procedure for new versions of the product. The IOFCvvvM, IOFCvvvA, IOFCvvvU,and IOFCvvvP modules can just be copied into the link list, since their names are guaranteed not to conflict with the production load module names for previous versions.

Then, by using an alternate transaction code and a renamed copy of the new IOFCIC module, you can completely test the new version. The install library contains jobs that are specifically designed to install new releases for test in this manner (See Chapter 6 for more details).

19. Entering MVS and JES2 Commands

For a user with OPERATOR privileges, to enter a JES2 command from any IOF screen, enter "**$**" followed by the command:

```
$DA
```

To enter an MVS command, enter "**#**" followed by the command:

```
#D  T
```

Entering an MVS or JES2 command under IOF will automatically take you to the IOF Extended MCS console. See Chapter 13 of the ***IOF User Guide*** for more information about IOF Extended MCS console support. Each user can enter the **AUTOCON OFF** command to disable automatic console. **AUTOCON ON** re-enables automatic console.

20. Profile Management Features

The Profile Data Set

Profile information for IOF/CICS users is stored in a standard MVS partitioned data set. This data set must have a DD statement in the CICS startup JCL (see option member B74PRFDD) but does not require an entry in the CICS File Control Table. Profiles are stored as individual partitioned data set members. The member name for a user's profile is generally the same as his or her CICS userid (see option member A35USRID).

A user may or may not be required to have a profile member. If option member B76PRFRQ specifies that profiles are required, a profile must be created before a user can use IOF. If profiles are not required, a user can use IOF without an existing profile member. IOF will automatically create one, but only if the user sets a profile variable to a value different from the installation default value.

If you do not require profiles, very little effort will be required to maintain the profile data set. In fact, the only profiles you will explicitly need to add are those for systems personnel who need to be defined with ACCOUNT authority in order to use the profile manager, and those for any systems programmers who require OPERATOR authority. IOF will automatically create profile members as needed for your normal users.

In addition to containing a user's PF key definitions and other IOF related variables, the user's profile member also contains much of the information that would be in the SYS1.UADS data set for a TSO user. This includes such data as the user's default data set prefix and unit name, as well as any special attributes such as ACCOUNT, OPERATOR, or MOUNT.

The Profile Manager Panel

Enter "PM" on the *IOF Option Menu* to invoke the *Profile Manager Panel*. This option will appear on the *Option Menu* only if you have ACCOUNT authority (see Chapter 2 for information about using the "/$MASTER" option to define the first id with ACCOUNT authority). From the *Profile Manager Panel* an authorized user can add, delete, or update profile information for other users.

Setting Profile Values

Under the *Profile Manager Panel* there is always a current set of working profile values. When the panel is first invoked, they are set to the default profile values for your installation. There are several ways under the *Profile Manager Panel* to modify the working profile values.

Of course, any changes to the working profile values here will have no effect on the profile values for the user who is using the *Profile Manager Panel*. These working profile values will only be used to add profile information for new users or to update profile information for other existing users.

When an ADD or SAVE function is performed, the current profile values in the working profile will be saved for the indicated userid, but the working profile values will not be reset. This means that if you want to add several users with the same profile values, you can set up the profile values and enter multiple ADD commands for the individual users.

If you wish, you can use the normal profile panels to modify the working profile values. Enter "P" on the *Profile Manager Panel* to invoke the *Profile Option Menu*, from which you can select the appropriate profile panels to make your desired changes. When you make changes on these panels, you will not be changing your own profile values. You will be updating the current working profile values for the *Profile Manager Panel*.

You also can specify that you want the set of working profile values to be the same as those for an existing user. To do this just enter **M** in the command field of the *Profile Manager Panel*, enter the userid whose profile you want to copy in the USERID field, and press **ENTER**. When the **M** command completes, the working profile will match the profile of the specified user.

You optionally can enter **MODEL userid** (or **M userid**) in the command area to accomplish the same function.

You can also set profile values by entering a **SETPVAR** command with the desired variable name and value (SETPVAR variable-name value). See Chapter 6 of the ***IOF User's Guide*** for a description of the **SETPVAR** command. In addition to the variables described in that document, the following variables are supported on the **SETPVAR** command for authorized users only:

DFLOPER. Specifies (YES/NO) if the user should be granted IOF operator status. For example **SETPVAR DFLOPER YES** will give the user OPERATOR.

DFLACCT. Specifies (YES/NO) if the user should be allowed to make changes in other user's profile information.

DFLMOUNT. Specifies (YES/NO) if the user should be allowed to mount offline devices in response to an **SD** or **SAVEINDX** command. You will rarely, if ever, want to specify YES.

DFLPRFX. Specifies the default dsname prefix to be used when a user creates a new data set and the data set name is not enclosed in quotes. You can specify a 1-7 character constant or one of several "special" values described in option member A36PREFX. For example, **SETPVAR DFLPRFX @USERID** will cause the user's userid to be used as the prefix value.

DFLGROUP. Specifies the 1-8 character esoteric group name (unit) for new data sets created in response to **SD** or **SI** commands for this user. For example, SYSDA or DISK.

To reset the working profile values back to the system default for your installation enter the **RESET** command.

You can specify that a user's userid is to be mapped to another userid for use under IOF by filling in the desired alternate userid in the ALTUSER field of the *Profile Manager Panel* when the **ADD** or **SAVE** command is entered. This feature is essentially obsolete after CICS release 1.7 which was the first CICS release to support real userids (see option member A35USRID for details).

Adding a New Profile Member

After you have defined the desired profile values using the methods described above, you can add a new userid profile member having those values by either:

Entering **A** in the command field and the userid in the USERID field,

or

Entering **ADD userid** n the command field.

Deleting an Existing Profile Member

To delete an existing profile member you can either:

Enter **D** in the command field and the userid in the USERID field,

or

Enter **DELETE userid** in the command field.

Updating an Existing Profile Member

To update an existing profile member you would normally set the current working profile to the profile values for the user by either:

Entering **F** in the command field and the userid in the USERID field,

or

Entering **FETCH userid** in the command field.

You can then use any of the methods described above to modify the profile values for the user. When you have made the desired changes to the user's profile information, you can update the saved profile data for the user by entering **SAVE** in the command area.

If you have previously entered a **FETCH** command to load the user's old profile information, the user's userid should still be displayed in the USERID field. If the userid is not displayed in the userid field then you should either:

Enter "SAVE" in the command area and the userid in the USERID field,

or

Enter "SAVE userid" in the command field.

21. Local Data Set Name Prefixing

If a data set name is specified in the **SD** command (or on the SD interface panel) without enclosing it in apostrophes, IOF will prefix the name with the user's current profile prefix character string. This prefixing is done in the source module DSNQUAL, which is included in the IOF source library.

Options member B54SNPDS describes other options that can be used to control snap data set names.

22. Sample IOF Modifications

The sample modifications in the IOF sample modifications library were contributed by IOF customers and have not been installed or tested by our technical support staff. They are provided as potentially useful indications of how you might approach the job of modifying IOF to accomplish certain installation objectives.

The member $INDEX contains a list and brief description of the modifications contained in the library.

23. IOF/CICS Master Terminal Command

This chapter describes how to use the IOF/CICS master terminal command (transaction IOMT). The IOF/CICS master terminal command is generally not required for the normal operation of IOF in most installations, although it may prove to be useful in some situations.

The master terminal command functions are performed by program IOFCMT, which is invoked by transaction code "IOMT". The syntax for the IOMT transaction is as follows:

IOMT INQ	Display number of active IOF users
IOMT SHUT	Terminate (abend) all active IOF users
IOMT TSPURGE	Purge the IOF temporary storage queue
IOMT OSLOAD=x	Issue a MVS LOAD macro for module "x" from the MVS link list
IOMT OSDELETE=x	Issue a MVS DELETE macro for module "x"
IOMT CILOAD=x	Issue an **EXEC CICS LOAD HOLD** command for module "x"
IOMT CIDELETE=x	Issue an **EXEC CICS RELEASE** command for module "x"

The IOMT transaction can be entered from a CICS terminal or can be entered from a MVS console using the MVS modify command. For example:

```
F cicsname,'IOMT INQ'
```

The IOF/CICS master terminal command can also be invoked from a command level application program by invoking program IOFCMT and passing a COMMAREA containing an IOMT transaction just as it would have been entered from the terminal. In this case any command responses or error messages will be sent to the IOF transient data destination specified in option member B15TRDTA. To route the responses and error messages to the MVS console, append "MSG=WTO" to the command being passed. For example:

```
EXEC CICS LINK PROGRAM('IOFCMT') COMMAREA('IOMT
INQ,MSG=WTO')
```

An example program which loads IOF/CICS modules into the CICS address space at CICS initialization time can be found in the product sample modifications library.

24. Functional Comparison of IOF/CICS and IOF/TSO

IOF/CICS appears to the end user to be almost identical to IOF running as a TSO command processor at the TSO READY level. There are, however, differences that result from limitations inherent in the CICS environment. These are described in the following paragraphs.

Batch TMP

IOF/CICS runs only as a terminal oriented task under CICS. It cannot be run as a batch job or started task using the batch terminal monitor program facilities of TSO.

CLISTs, REXX EXECs, and ISPF Dialogs

Clists such as INDEXASM and INDEXDMP and REXX execs such as IOXSETUP which are provided with the TSO version of IOF cannot be executed from IOF/CICS. Likewise, IOF global commands such as **DQ**, **DG**, and **FORMAT** which invoke TSO clists, and commands that invoke the *Book Manager Read* program cannot be used under CICS. Functions that invoke ISPF dialogs from IOF such as the job archival and retrieval functions are not allowed under CICS.

TSO Attention

CICS does not provide a facility to interrupt the execution of a running transaction. Consequently, IOF/CICS provides an option that allows the installation to specify the maximum elapsed time that the user's keyboard can remain locked before IOF will prompt the user for a simulated attention. For example, if the user attempts to use the **FIND** command to find a non-existent string in a very large sysout data set, the user will be prompted after this time interval has expired to ask if processing should be continued or an attention condition should be simulated.

SYSLOG/OPERLOG

Since IOF/CICS does not support clists and cannot be run using the batch TMP, it cannot be used to create an index data set for use by the **LOG** command. However, the **LOG** command can be used to browse

syslog/operlog without an index. Further, if the installation has IOF/TSO, the index created for IOF/TSO can be used to view the log from IOF/CICS by specifying INDEX=YES in options member B30SLAM.

25. CICS External Security Considerations

IOF provides the ability to display and update JES2 resources and certain disk data sets. You can use your security system to control IOF access to these resources under CICS. In some cases there are special considerations for using RACF, Top Secret, or ACF2 with IOF under CICS.

This document assumes that you are running CICS 1.7 or later. Many IOF access control facilities are available for older CICS releases. Please call us if you have a CICS system prior to 1.7 and would like to use your security system to control IOF access.

IOF Profile Data Set

The userid under which the CICS address space is running will need update access to the IOF profile data set.

Target Data Sets for SNAP and SAVEINDX

IOF allows the user to copy sysout data into an MVS disk data set. Also, a user can save the internal index for a browse session into a disk data set with the **SAVEINDX** command.

The A34DSWRT options member allows you to specify how you want to control the users' ability to write into these data sets:

> **DSWRITE=NEVER** says that you do not want to allow users to snap to disk data sets or use the **SAVEINDX** command.
>
> **DSWRITE=UNQUOTED** says that you only want users to be able snap or issue **SAVEINDX** commands to data sets that are specified without quotes. These data sets will be prefixed by the users own prefix (see options member A36PREFX).
>
> **DSWRITE=RACF** says that you want to let your security system decide which data sets users are allowed to write into.

If you specify DSWRITE=RACF, there are considerations that depend on which security system you are running:

For RACF, use the default versions of options A35USRID and C81CXSEC. You do not need to specify RACF in the A60ACF option since that option does not affect access to disk data sets.

For Top Secret, use the default versions of A35USRID and C81CXSEC just like for RACF. In addition, you must modify the TSS facility definition for any CICS system in which IOF will be used as follows. For all releases of Top Secret, the CICS facility definition must specify the RES option. For Top Secret 4.2 and earlier, the TENV option must also be specified.

For ACF2, also use the default versions of A35USRID and C81CXSEC. In addition, your ACF2 GSO OPTS record must specify SAF and your ACF2 GSO SAFPROT record should include the following entries:

```
SAFPROT.IOFCICS1 SUBSYS(SVC019) CNTLPTS(IOFC-) CLASSES(-)
SAFPROT.IOFCICS2 SUBSYS(SVC099) CNTLPTS(IOFC-) CLASSES(-)
SAFPROT.IOFCICS3 SUBSYS(IOFC-)  CNTLPTS(IOFC-) CLASSES(-)
```

Make sure that the userid under which CICS executes does not specify NO-SAF. This will override the GSO OPTS record and the user will get the following message when trying to use IOF:

```
RACROUTE VERIFY CREATE failed for userid: xxxxxxx
SAF and ESM return and reason codes were: 04000000
```

Controlling IOF Resources

To control IOF resources (jobs, JES2 devices, etc.) with your security system you must specify which security system in options member A60ACF. See Chapter 26 for more information about how to control specific resources.

When controlling IOF resources with your security system, there are considerations that depend on which security system you are using:

For **RACF**, specify RACF in the A60ACF options member and use the default versions of A35USRID and C81CXSEC.

For **Top Secret**, specify TSS in the A60ACF option and use the default versions of A35USRID and C81CXSEC. Modify the TSS facility definition for any CICS system in which IOF will be used as follows. For all releases of Top Secret, the CICS facility definition must specify the RES option. For Top Secret 4.2 and earlier, the TENV option must also be specified.

For **ACF2**, specify ACF2 in the A60ACF option and use the default versions of A35USRID and C81CXSEC. You must also make the ACF2 options changes described in the "Data Sets" section above. Remember that if you do change C81CXSEC after you initially install IOF, you must re-run the M10INIT job and then do a full generation of the product (see Chapter 3).

26. Access Control Reference

Introduction

Chapter 9, ***Access Control Overview***, describes how to use IOF facilities to control access to IOF resources. This chapter contains a more detailed look at the underlying structure of IOF access control.

By default IOF allows most users to control only their own jobs. End users cannot control devices, browse the system log, or use the *IOF System Monitor* unless specifically authorized. Users with TSO operator authority are allowed to browse and control all jobs and devices in the system and to use all IOF facilities.

IOF access control facilities let you change these default rules to meet the requirements of your installation. End users easily can be permitted to browse and control jobs they don't own or to manage specific devices. Operator users can be prevented from browsing sensitive jobs such as the payroll. These changes can be made by parameter changes or by interfacing IOF to your host security system.

IOF allows you to control access to jobs, output groups, sysout data sets, JES2 devices, system commands, and other systems in a sysplex. Access rules are defined through several IOF options which are members of the IOF options library. The detailed description of these options is contained in the options library. This chapter explains how the various options fit together and gives specific access control examples.

Access Control Options Members

The following members of the IOF Options Library are used to define the access control rules for your installation:

- **A35USRID**. This option allows you to specify the association between CICS users and IOF user ids.
- **A40SCOPE**. This member defines the default job ownership rule for your system.
- **A60ACF**. Specifies which security system you have (RACF, ACF2, or Top Secret) and whether operators and started tasks should be allowed access without requiring rules in the security system.
- **B21$DOC**. This member contains the documentation for the B21ACESS member.

- **B21ACESS**. This member describes the level of IOF access required to perform each of the functions defined by IOF.
- **B23$DOC**. This member contains the documentation for the B23ALLOW member.
- **B23ALLOW**. This member allows you to assign IOF users to access control groups. It also allows you to control which users (or groups) are allowed to do which IOF functions.
- **B24ACFDF**. Use this member to select the types of access that you want to protect with your security system. You also specify the high level prefix for all IOF security system resource names.
- **B25DVGRP**. This member groups JES2 device functions and parms together into smaller units that can be referenced more easily in B21ACESS.

Defining Default Job Ownership

The A40SCOPE option defines the default job ownership rule for your site. The selection that you make in this options member will heavily affect all other access control options for IOF.

In A40SCOPE you indicate which jobs are to be associated with an individual user. Jobs can be associated based on job name or based on job owner.

Job owner is much more attractive, since it removes all restrictions on the names that users can assign to their jobs. But, it does require that an owning userid be assigned to each job by JES2 and stored in the JES2 JQE control block for the job.

If you want job ownership to be based on the owning userid, specify:

```
USSCOPE  OWNER,'/U'
```

Otherwise, you would specify A40SCOPE as:

```
USSCOPE  JOBNAME,'/U*'
```

which will cause IOF to assume that a job is associated with a user if the job's name begins with the user's userid.

Read the A40SCOPE options member and be sure you understand the choices you can make. Select your USSCOPE option, and keep it in mind as you continue.

Defining IOF User Groups

IOF is shipped with three groups defined:

- **OPERATOR**. All users with the operator attribute. (UADS=OPE)
- **STCGROUP**. All started tasks. (ASIDTYP=STC)

- **ENDUSER**. All other users. (No qualification parms)

Options member A60ACF controls the functions that are available to members of the default operator and started task groups. End users are allowed to review and control only the jobs they submitted. You easily can change the authority that the default groups have and can define as many IOF groups as you need.

If you have converted the ISFPARMS data set for IBM's SDSF product, you will have one IOF group for each ISFGRP macro in the ISFPARMS data set.

Use GROUP macros in options member B23ALLOW to define your IOF groups. See options member B23$DOC for a complete description of the GROUP macro.

Each user is assigned to the first group for which they qualify, so the order of GROUP macros in B23ALLOW is important. The most restrictive group macros should appear first in B23ALLOW and the least restrictive ones later. Users who do not qualify for any IOF group will not be allowed to use IOF.

One or more of the qualification parms described in the table below can be included on the GROUP macro to define the users that qualify for the group. If there is no qualification parm on a GROUP macro, all users qualify for that group.

Qualify User	Qualify STRLIST	Exclude User	Exclude STRLIST	Description
ID	IDLST	XID	XIDLST	One or more explicit user ids
PROC	PROCLST	XPROC	XPROCLS	One or more TSO logon procedure names
TERM	TERMLST	XTERM	XTERMLS	One or more terminal names
ACCT	ACCTLST	XACCT	XACCTLS	One or more account numbers
ACFGP	ACFGLST	XACFGP	XACFGLS	One or more RACF group names
ACFLG	ACFLGLS	XACFLG	XACFLGL	One or more RACF connect groups
AC2UID	AC2UIDL	XAC2UID	XAC2UIL	One or more ACF2 userid strings
SES1 SES2	SES1LST SES2LST	XSES1 XSES2	XSES1LS XSES2LA	One or more session attributes named in source member ATTRBASE. Specify the parm as shown: SESn=(attrname,(attrval,..))
UADS				UADS attr: OPERATOR, MOUNT, ACCOUNT, JCL
ASIDTYP				Address space type: JOB, TSU, STC

User Qualification Parms

For example, the parm "ID=(HR097A,HR177B,HR9*)" on a GROUP macro qualifies userids "HR097A", "HR177B", and all ids beginning "HR9" for the group. The parm "IDLST=PAYLIST" qualifies all ids listed in the STRLIST macro at label "PAYLIST". The parms "UADS=OPER,XID=OPNEW" qualify all users with operator authority except "OPNEW".

IOF Group Features

Several IOF features are controlled by parameters on the GROUP macro. The table below lists the parm names and describes the group feature each parm controls.

Parm Name	Values	Description
FINDLIM	(max,default)	Maximum allowable and default FINDLIM
EXCLTSO	YES/NO	YES means exclude the active TSO session from display unless it has output. NO means always display active session.
MINPAUS	seconds/NONE	Minimum pause or refresh time allowed in seconds
EXTEND	YES/NO/DEFAULT	EXTEND command is allowed, not allowed, or used by default
INITCMD	INITCMD/INITMENU	*Job List Menu* or *IOF Option Menu* on initial IOF entry
ACTION	route1,route2,ALL/ NONE	WTOR route codes that will be displayed by default at the bottom of the system log display
MONITOR	opt1,opt2.../NONE	System monitor default options or NONE
SYSID	systemid	Default syslog system id
FORMATS	(sect1,sect2...)	Alternate display section format names
CMDTYPE	(type1,type2...)	Global command types
PANEL	OPTOPT/OPTUS1/OPTUS2/ OPTUS3/OPTSDS	*IOF Option Menu* name
ALLOW	(alow1,alow2...)	ALLOW macro names that apply to this group
DFSCOPE	USER/GROUP/ALL	Default scope
MXSCOPE	USER/GROUP/ALL	Maximum allowable scope
USSCOPE	(JOBNAME,str1...str8) (OWNER,str1...str8)	User scope definition. Defaults to the value set in the A4OSCOPE option.
GRSCOPE	(JOBNAME,str1...str8) (OWNER,str1...str8)	Group scope definition
INPCMD	YES/NO	Allow use of the Input command to display input data sets on *Job Summary Menu*
QOPT	YES/NO	"Q" option allowed
DISPLAY	(refresh,display)	.01 seconds minimum display refresh/status interval
CONSOLE	YES/NO	CONSOLE command allowed
DRCMD	YES/NO	Display replies (DR) command allowed
STR1,STR2, STR3,STR4	string (string,beg,length)	String 1 through string 4 definitions. "/U" means users userid. "*" is wild card terminator. "+" wild card position.
AUTH	(one or more options)	Specific options to be displayed on the *IOF Option Menu*
DOCLEVL	ENDUSER/OPERATOR/ ADMIN	Level of commands to be documented on the MORE command

Parm Name	Values	Description
AUTHADD	(one or more options)	Specific options to be added to the default options displayed on the *IOF Option Menu*
AUTHREM	(one or more options)	Specific options to be removed from the default options displayed on the *IOF Option Menu*
MENUGLOB	YES/NO	YES means that IOF Options are honored on any IOF panel.
OCMD	GROUPS/JOBS	Defines whether the "O" option should display groups or jobs.

GROUP Macro Feature Parms

The default GROUP macros in options member B23ALLOW are surrounded by comments that describe the functions being defined for each group.

IOF Resources

IOF controls access to jobs, output groups, sysout data sets, commands, systems and devices. Specific users or groups of users can be allowed to look at or modify any of these five basic resource types. The names of these resource types are:

- JOBS
- GROUPS
- SYSOUTS
- DEVICES
- COMMANDS
- SYSTEMS
- PROCESS
- THREADS
- ENCLAVES
- SCHENV
- SCHRES
- CHECKS
- JOBCLASS
- VOLUMES
- NODES

You will use these resource names when permitting users access to IOF resources.

IOF Resource Attributes

Each type of IOF resource has a set of attributes or characteristics. Access to a resource can be granted based on any one of these attributes. For

example, each job has a name and a print destination, and may have an owner, a notify id, and other characteristics.

IOF resource attributes are defined in source member ATTRBASE. You can define your own resource attributes by modifying options member B63ATTR. The following table lists many of the attributes that have already been defined. It shows both primitive and combined attributes. Combined attributes are simply a combination of two or more primitive attributes.

Resource Type	Attribute Name	Attribute Type	Description
JOBS	JOBCOMBO	Combined	OWNER.JOBNAME
	JOBNAME	Primitive	Name of the job
	OWNER	Primitive	Userid of the job owner
	NOTIFY	Primitive	Notify userid
	DEST	Primitive	Job level destination
	CLASS	Primitive	Job class
	JOBID	Primitive	Job id
	ACFGROUP	Primitive	RACF group of owner
	ACF2UID	Primitive	ACF2 userid string
	SUBUSER	Primitive	Submitter's userid
	SUBGROUP	Primitive	Submitter's group
	A19JBTYP	Primitive	Job type (BAT, STC, TSU)
	ACCT	Primitive	Account number
GROUPS	DEST	Primitive	Destination of group
	CLASS	Primitive	Class of group
	FORMS	Primitive	Forms of group
	WTRID	Primitive	External writer name of group
	MAILED	Primitive	Mail id of the group
	USC, FCB, …	Primitive	Other group parms
SYSOUTS	DEST	Primitive	Destination of the data set
	CLASS	Primitive	Sysout class of the data set
	FORMS	Primitive	Forms of the data set
	WTRID, FCB,USC, …	Primitive	Other sysout characteristics
	PDVDSKEY	Primitive	Data set key
	PDVDDNAM	Primitive	DDNAME
DEVICES	DEVCOMBO	Combined	DEVTYPE.DEVNAME

Resource Type	Attribute Name	Attribute Type	Description
	DEVNAME	Primitive	Device name
	DEVTYPE	Primitive	Generic device type
	DEST	Primitive	Devices associated with the destination
COMMANDS	CMDCOMBO	Combined	CMDTYPE.CMDNAME.CMDPARM1
	CMDTYPE	Primitive	Command type (MVS or JES)
	CMDNAME	Primitive	Command name
	CMDPARM1	Primitive	First positional parm
	COMMAND	Primitive	(for compatibility with prior releases)
SYSTEMS	SYSID	Primitive	System id (SYSID or Service name)
PROCESS	JOBNAME	Primitive	Process Jobname
	OWNER	Primitive	Process OWNER
	TYPE	Primitive	TSU, STC, BAT
THREADS	Currently none		
ENCLAVES	SSTYPE	Primitive	Subsystem type
	SUBSYS	Primitive	Subsystem name
	OWNERJOB	Primitive	Enclave owner jobname
SCHENV	NAME	Primitive	Environment name
SCHRES	NAME	Primitive	Resource name
CHECKS	OWNER	Primitive	Owner of check
	NAME	Primitive	Check name
	SYSNAME	Primitive	System name for check
	PROCNAME	Primitive	Procedure name for checker
	STCID	Primitive	Started task ID for checker
JOBCLASS	CLASS	Primitive	One character job class
VOLUMES	VOLUME	Primitive	Volume serial number (name) of the spool volume
NAME	NODES	Primitive	Node name

Resource Attributes by Resource Type

Session Attributes

The current IOF user's session has certain attributes that can be used for access control. When the user tries to access a job, these session attributes can be matched against the same (or different) attributes of job.

Attributes	Description
USERID	User's userid
ASCBTYP	Type of address space, JOB, STC or TSU
ACCT	User's account number
XEQSYSID	Execution system id of the JES2 system
ACFGROUP	RACF group name
ACF2UID	ACF2 UID String

Session Attributes

Session attributes can be specified in the STR1, ..., STR8 parms of ALLOW macros to set up comparisons between an attribute of a job and the same attribute of the IOF user's session. See examples 7 and 8 in the ***Allow Macro Examples*** section below for more details.

IOF Access Levels

IOF defines four levels of display access and four independent levels of update access for each of the six resource types. Display access is required to see, browse or copy an IOF resource. Update access is required to modify an IOF resource.

Because display access is completely independent of update access, it is possible to permit users to "look but not touch". For example, you may wish to let users browse certain jobs without granting permission to cancel or modify the jobs. Conversely, you may want to let some operators route and cancel certain jobs under their control without granting them permission to browse the output. Both these requirements are easy to implement because IOF provides independent display and update access.

Options member B21ACESS defines which IOF functions are available at each level of access. Options member B21$DOC describes the macros that are used in B21ACESS. The table below shows the major IOF functions associated with each of the levels of IOF access in the default B21ACESS options member, and is somewhat easier to read than the B21ACESS member.

Resource Type	Level	Display Functions	Update Functions
Jobs	1	Display job on *Job List Menu*	
	2	Select job for review	Cancel, route, release held ds, modify dest/class/sysid
	3		Hold/release/restart job, modify class/priority
	4	Dump job control blocks	Set independent mods, modify performance group

Resource Type	Level	Display Functions	Update Functions
Groups	1	Display on Menu	
	2	Select group for review	Cancel, modify class/dest/forms/ pagedef/address…
	3		Hold/release group, modify wtrid/ priority/prmode
	4		
Sysouts	1	Display on *IOF Job Summary*	
	2		Cancel, modify class/dest/forms/ pagedef/address/wtrid/linect/prmode/ notify/usrlib …
	3	Browse and snap all data sets	
	4		
Devices	1	Display on *Device List Menu*	Start/drain/interrupt/restart/ backspace/set forms/ucs/fcb …
	2		Cancel, set class/flash/spacing/xeq node/sysid … Set offload dsname, unit, volser and type …
	3		Set dest/prmode/command authority, trace, disc intvl …
	4		Set wtrid/work select, initiator class, FSS name, autologon …
Commands	1		
	2		Issue DR command
	3		
	4		Issue JES2 and MVS commands
Systems	1	Display MAS on menu	Invoke server (functions on server are controlled by the server CPU's IOF)
	2	Select detail MAS display	
	3	Display Operlog	
	4		Start, stop, restart, reset, modify MAS parms Delete Operlog data
Enclaves	1	Display Enclaves on the menu	
	2		
	3		
	4		Quiese, resume, reset, set Service Class
Process	1	Display Process on the menu	
	2		
	3		
	4		Kill Process
Schenv	1	Display environment on the menu	

Resource Type	Level	Display Functions	Update Functions
	2		
	3		
	4	Display jobs and resources using the environment	
Schres	1	Display resource on the menu	
	2		
	3		
	4		Set the state and systems of the resource
Checks	1	Display check on the menu	
	2	Display check detail information	
	3		
	4	Browse and edit a check report	Activate, refresh, deactivate, delete, force, run, update and set check characteristics
Jobclass	1	Display class on menu	
	2	Select detailed display	
	3	Display JOBS on class	
	4		Modify all class attributes
Volumes	1	Display volume on men	
	2	Select detail display, DL	
	3	Display JOBS on volume	
	4		Set, drain, start, halt, sysaff
Nodes	1	Display node on menu	
	2	Select detailed display, DL, DC, DP	
	3		
	4		Set, start, all attributes

Standard Access Control Table

Granting a particular level of access will allow all functions with level numbers less than or equal to the granted level. So, granting level 3 display access also allows the display functions defined at level 1 and level 2.

A missing level number in the table above means that no new functions are introduced at that level. For example, no level 1 update functions are defined for jobs.

You can move functions and/or operands from one level of access to another by modifying options member B21ACESS. Member B21EX01 in the sample mods (SAMPMOD) data set shows an example of moving the input class and priority operands from level 3 update access to level 2. It is also possible to define additional access control tables if four levels of access are not sufficient.

Granting Access to IOF Functions

ALLOW macros in options member B23ALLOW define exactly which level of access is permitted to which type of resource under what conditions. No IOF function can be done unless it is permitted by an ALLOW macro.

Each ALLOW macro can apply to one or more specific users or groups of users. There are several ways of defining who is granted access by an ALLOW macro. The ALLOW and ALOWLST parameters on a GROUP macro point to ALLOW macros that are to apply to the group. Any of the user qualification parms described below can be specified on ALLOW macros to indicate which users are being permitted access. The ACF parameter on an ALLOW macro specifies that access is being granted through your security system.

Qualify User	Qualify STRLIST	Exclude User	Exclude STRLIST	Description
ID	IDLST	XID	XIDLST	One or more explicit user ids
GROUP	GRPLST	XGROUP	XGRLST	One or more IOF group names
PROC	PROCLST	XPROC	XPROCLS	One or more TSO logon procedure names
TERM	TERMLST	XTERM	XTERMLS	One or more terminal names
ACCT	ACCTLST	XACCT	XACCTLS	One or more account numbers
ACFGP	ACFGLST	XACFGP	XACFGLS	One or more RACF group names
ACFLG	ACFLGLS	XACFLG	XACFLGL	One or more RACF connect groups
AC2UID	AC2UIDL	XAC2UID	XAC2UIL	One or more ACF2 userid strings
SES1 SES2	SES1LST SES2LST	XSES1 XSES2	XSES1LS XSES2LA	One or more session attributes named in source member ATTRBASE. Specify the parm as shown: SESn=(attrname,(attrval,..))
UADS				UADS attr: OPERATOR, MOUNT, ACCOUNT, JCL
ASIDTYP				Address space type: JOB, TSU, STC

User Qualification Parms

For example, an ALLOW macro that has the parm "ID=('SYS*','OPR*') applies to all userids that begin "SYS" or "OPR". The parm "ACFLG=PAYCHECK" causes the ALLOW macro to apply to users that can connect to the "PAYCHECK" RACF group. "XGROUP=ENDUSER" means that the ALLOW macro applies to all IOF groups except the ENDUSER group.

Now, consider the following example ALLOW macro:

```
ALLOW   3,2,JOBS,JOBNAME,'ABC*',ID=('ABCX*','ABCY*')
```

Each ALLOW macro reads like a sentence. This one says "allow level 3 display and level 2 update access to all jobs with a job name beginning with ABC to all users whose userids begin with ABCX or ABCY".

The “ID=” parm of this ALLOW macro indicates to whom access is being granted. You also can specify that an ALLOW macro applies to every member of an IOF group by pointing to the ALLOW macro directly from a GROUP macro:

```
DEVGROUP   GROUP   ID=('ABCX*','ABCY*'),ALLOW=DEVJOBS
DEVJOBS    ALLOW   3,2,JOBS,JOBNAME,'ABC*'
```

All IOF access is defined through ALLOW macros in the B23ALLOW options member. More information about ALLOW macro features is contained in options member B23$DOC.

ALLOW Macro Description

Syntax

```
label  ALLOW
         dlev                      display level
         ,ulev                     update level
         ,res-type                 resource type
         ,res-attr[(col,len)]      resource attribute
         ,| match-str          |   match strings
          | STRLST=listname     |   lists of strings
          | NOT=match-str       |   match strings
          | NOTLST=listname     |   lists of strings
         ,| [user-qual]         |   user qualifier
          | [ACF=| GLOBAL |]   |   security system
          |      | USER   |    |   global/user levels
         [,TABLE=access-tbl]       access table name
         [,STR1,...,STR8=(session-attr[,(col,len)])]
         [,IFALSO=(atrc-nam#1,...,atrc-nam#n)]
```

label. The name of the ALLOW macro. This name can be used in the ALLOW operand of a GROUP macro to associate this ALLOW macro with the group. A label is useful for tracing purposes even if you do not reference it.

dlev. The level of display access being allowed, with a value from 0 to 4. If the ACF parameter is specified, this is the maximum display level that can be granted by your security system.

ulev. The level of update access being allowed, with a value of 0 to 4. If the ACF parameter is specified, this is the maximum update level that can be granted by your security system.

res-type. The type of resource to which access is being allowed. Valid resource types are JOBS, GROUPS, SYSOUTS, DEVICES, COMMANDS, SYSTEMS, PROCESS, THREADS, AND ENCLAVES.

res-attr. The resource attribute name. This specifies the characteristic of the resource that is to be compared against the match-strings to determine if this ALLOW macro is applicable. For example, if res-type were JOBS, res-attr might be JOBNAME or OWNER. For valid resource attributes for each resource type, see the table in the above section, ***IOF Access Levels***.

Note that a value of an asterisk (*****) can be specified for the resource attribute name to indicate that this ALLOW macro is applicable to all resources of the type named in res-type. When this form is used, the res-attr and match-str parameters are not used.

res-attr(col,len). Specifies a substring of the resource attribute. This would normally only be used in conjunction with the STR1, ..., STR8 parms. See Example 8 in the section below, ***ALLOW Macro Examples***.

match-str. The set of pattern match strings that are to be compared against the resource characteristic named in res-attr above to determine if this ALLOW macro is applicable. If an asterisk (*****) is specified for res-attr, this parameter should be omitted. Generic names can be specified using the plus (**+**) as a one character wild card, and the asterisk (*****) as a wild card terminator. More than one name can be specified if enclosed within parentheses.

STRLST=listname. An alternate way to specify the match-strings. "listname" is the name of a STRLIST macro that describes a list of match strings.

NOT=match-str. A set of pattern match strings that must not match the resource attribute in order for the ALLOW macro to be applicable.

NOTLST=listname. An alternate way to specify the NOT= strings. "listname" is the name of a STRLIST macro that describes a list of match strings.

user-qual. Specifies which users this ALLOW macro applies to. See the table in the above section, ***Granting Access to IOF Functions***,

for a description of all the user qualification parms that can be used on an ALLOW macro.

ACF=GLOBAL. Specifies that this is a pattern ALLOW macro that describes a profile (rule) defined to your security system that should be checked to determine if this user should be allowed to perform the function they are attempting. GLOBAL implies that the rule is defined at the system level and has the prefix specified in the B24ACFDF options member. System level rules are controlled by the system security administrator and not by individual users. See the below section, ***Using Your Security System to Control IOF Access***, for more information about defining security system rules that correspond to IOF ALLOW macros.

ACF=USER. Specifies that this is a pattern ALLOW macro that describes a profile (rule) defined to your security system that should be checked to determine if this user should be allowed to perform the function they are attempting. USER implies that the rule is defined at the user level and has a user's userid as a prefix. User level rules are controlled by individual users. See the below section, ***Using Your Security System to Control IOF Access***, for more information about defining security system rules that correspond to IOF ALLOW macros.

TABLE=access-table. Specifies the name of an access table in options member B21ACESS. This would never need to be specified unless you have defined additional access tables in B21ACESS. The name of the default access table is STANDARD.

STR1=session-attr. Defines the /1 insertion string to be the value of the named session attribute. See Example 7 in the section below, ***ALLOW Macro Examples***..

STR1=(session-attr,col,len). Defines the /1 insertion string to be a substring of the named session attribute. See Example 8 in the section below, ***ALLOW Macro Examples***..

STR2=, ..., STR8=. Define the /2, ..., /8 insertion strings.

IFALSO=(atrc-nam#1, ..., atrc-nam#n). Names one or more ATTRCHK macros that define additional conditions that must be met in order for the ALLOW (or LIMIT) macro to be honored.

ALLOW Macro Examples

Example 1. Let all users browse the system log by granting level 3 display access to jobname "SYSLOG" to all userids.

```
ALLOW  3,0,JOBS,JOBNAME,'SYSLOG',ID=*
```

Example 2. Let userids that begin "OPER" have "SYS" in positions 3 through 5, or begin "PR" and have "CL" in positions 4 and 5 browse the log, jcl and messages data sets of all jobs in the system based on any selection criteria. Also, let them cancel all jobs and modify all job attributes. Do not permit browsing the regular sysout data sets of the jobs.

```
ALLOW  2,4,JOBS,*,ID=('OPER*','++SYS*','PR+CL*')
```

This can also be accomplished by:

```
       ALLOW    2,4,JOBS,*,IDLST=LN1
LN1    STRLIST  'OPER*','++SYS*','PR+CL*'
```

Example 3. Define userid "RSAM" as the operator of remote 18. Do not allow RSAM to modify the destination, work select, and other systems type parameters of the printer.

```
ALLOW  2,2,DEVICES,DEVNAME,'R18.*',ID='RSAM'
```

Example 4. Allow everyone in the 'OPERATOR' and 'SYSPGMR' groups to fully control all devices.

```
ALLOW  4,4,DEVICES,*,GROUP=(OPERATOR,SYSPGMR)
```

Example 5. Allow all users connected to the MSTRCTL RACF group to browse and modify most characteristics of all jobs with a notify id of 'SYSCTL'.

```
ALLOW  3,2,JOBS,NOTIFY,'SYSCTL',ACFGP=MSTRCTL
```

Example 6. Allow all users who are connected to the "LOCOPER" RACF group to control all sysout groups and devices that are routed to or associated with the "LOCAL" destination.

```
ALLOW  4,4,GROUPS,DEST,'LOCAL',ACFLG=LOCOPER
ALLOW  4,4,DEVICES,DEST,'LOCAL',ACFLG=LOCOPER
```

Example 7. Allow all users to control jobs in their own RACF group.

```
ALLOW  3,2,JOBS,ACFGROUP,'/1',ID=*,STR1=ACFGROUP
```

Example 8. Allow all users to control a job if the seventh and eighth characters of the job's ACF UID string match the seventh and eighth characters of the IOF user's ACF2 UID string.

```
       ALLOW  3,2,JOBS,ACF2UID(7,2),'/1',ID=*,STR1=
  (ACF2UID,7,2)
```

Example 9. Allow everyone except the IOF ENDUSER group to display the MAS and invoke the AT command on all systems.

```
ALLOW  2,1,SYSTEMS,SYSID,*,XGROUP=ENDUSER
```

Example 10. Allow ENDUSER group members to invoke the AT command for sysid IPO2. Functions available during the server session are controlled by the IOF on the server CPU.

```
ALLOW  0,1,SYSTEM,SYSID,IPO2,GROUP=ENDUSER
```

Example 11. Allow all users to browse all sysout data sets with a ddname of SYSPRINT and sysout class of "J".

```
ALLOW  2,0,JOBS,*,ID=*
ALLOW  3,0,SYSOUTS,DDNAME,SYSPRINT,ID=*,
IFALSO=SOCLASSJ SOCLASSJ  ATTRCHK  SYSOUTS,CLASS,J
```

Example 12. Prevent all access to sysout classes "C" and "R" for jobnames beginning "HR" to everyone except userids beginning "HRM".

```
LIMIT 1,0,SYSOUTS,JOBNAME,HR*,XID=HRM*,IFALSO=
PAYCLASS   PAYCLAS  ATTRCHK  SYSOUTS,CLASS,(C,R)
```

See SAMPMOD library members B23EX02 and B23EXSES for additional examples of ALLOW macros.

Limiting Access with LIMIT Macros

The LIMIT macro has exactly the same parms as the ALLOW macro but limits access rather than allowing access. The level numbers on a LIMIT macro indicate the highest possible access that will be granted to the resource. For example:

```
LIMIT  2,2,JOBS,JOBNAME,'PAY*',ID=*
```

This macro sets a limit on the level of IOF access that can be allowed to any job whose name begins with "PAY". Since ID=* is specified, this limitation applies to all IOF users. This macro prevents all users from browsing any sysouts other than the log, JCL, and messages for all jobs with jobnames beginning "PAY".

A LIMIT macro that absolutely applies to all users is somewhat unusual in practical situations. Normally there are a few users for whom the limit should not be applied. The following LIMIT macro is an example:

```
LIMIT 1,0,SYSOUTS,CLASS,'P',XACFLG=PAYROLL
```

This macro prevents browse or modify of class "P" sysout data sets except by users who are connected to the PAYROLL RACF group. Class "P" data sets are only allowed to be displayed on the Job Summary display for all other users.

See SAMPMOD library member B23EXLIM for several additional examples of LIMIT macros.

Defining Multiple Attributes with the ATTRCHK Macro

The ATTRCHK macro is used to define additional criteria that must be satisfied in order for an ALLOW or LIMIT macro to be honored. ALLOW and LIMIT macros can point to one or more ATTRCHK macros with the IFALSO parm.

When the IFALSO parm is specified, the ALLOW/LIMIT macro will be used only if all of the specified ATTRCHK macros are satisfied.

Syntax

```
name     ATTRCHK      res-type
                      ,res-attr
                      ,|match-str      |
                       |STRLST=listname|
                       |NOT=match-str  |
                       |NOTLST=listname|
```

name. The name specified in the IFALSO parm of an ALLOW or LIMIT macro.

res-type, res-attr, match-str and listname. These have the same meanings as on the ALLOW and LIMIT macros. These parms define additional conditions that must be met. Note that the res-type for the ATTRCHK macro must match the res-type parm on the ALLOW or LIMIT macro that points to the ATTRCHK.

Special "CONTROL" Limit Attribute

A special "CONTROL" attribute for "JOBS", "GROUPS" and "SYSOUTS" combines several key attributes of these resources. The "CONTROL" attribute is built by combining the following attributes separated by a period (.).

Attribute	Length	Description
JOBID	8 characters	Jobid of the job
OWNER	1 to 7 characters	Owner of the job
JOBNAME	1 to 8 characters	Jobname of the job
JOBSTATE	5 to 7 characters	Job state: INPUT, RUNNING, OUTPUT
FUNCTION	4 to 7 characters	IOF validation function: MENU, SELECT, CANCEL, MODIFY, ROUTE, JRELEASE, DRELEASE, HOLD, DUMPCB, PRINT, RESTART, SNAP

The "CONTROL" attribute is especially useful in the LIMIT macro. For example, the following LIMIT macro prevents anyone except userids beginning "CO" from canceling any running started task with a jobname beginning "CICS":

```
LIMIT  0,0,JOBS,CONTROL,'S*.*.CICS*.RUNNING.CANCEL',XID=CO*
```

The LIMIT macro above only applies to running CICS jobs. It does not restrict cancel of CICS jobs on the output queue.

Building ALLOW and LIMIT Macros Using the ALLOW Command

The easiest way to build ALLOW and LIMIT macros is to use the IOF **ALLOW** command. From any IOF panel when running under ISPF, enter the ALLOW command to initiate a dialogue that takes you step-by-step through the process of building specific ALLOW and LIMIT macros.

You can save the new ALLOW and LIMIT macros if you wish, or you can simply review the macros that the dialogue builds in order to get a better understanding of how the generated macros work. No IOF options members are ever updated by the dialogue unless you explicitly specify the member name and verify the update.

If you have only the CICS version of IOF, you will have to edit the B23ALLOW options member to specify your ALLOW and LIMIT macros.

If you have both the TSO and CICS versions of IOF and you want to use the same access rules for both products, you can copy your IOF/TSO access control options members to your IOF/CICS options library. Make sure that both products are at the same release level, and then review options members A40SCOPE, A60ACF, B21ACESS, B23ALLOW, B24ACFDF, and B25DVGRP in your IOF/TSO options library. Copy the desired options members to your IOF/CICS options library and make whatever changes, if any, are required.

STRLIST and ADRLIST Macros

Both the GROUP and ALLOW macros have parms for which you may wish to specify lists of names, ids, or addresses. Sometimes, the same list needs to be specified several times.

The STRLIST macro allows you to define a list of match strings that can be referenced by ALLOW and GROUP macros.

Syntax

```
label   STRLIST 'str1','str2',...
```

label. The name used to point to this STRLIST macro.

'str1','str2',... The list of generic match strings.

The ADRLIST macro allows you to define a list of address pointers that can be referenced by GROUP and ALLOW macros.

Syntax

```
label   ADRLIST addr1,addr2,...
```

label. The name used to point to this ADRLIST macro.

addr1,addr2,... The list of address pointers.

Access to Sysout Data Sets

You will notice that the default B23ALLOW options member does not contain any ALLOW macros for the SYSOUTS resource type. This is because granting access to a job or output group also grants the same level of access to any sysouts in the job or output group.

You will need ALLOW macros for the SYSOUTS resource type only if you need to grant a higher level of access to some sysout data sets of a job than you grant to the job as a whole. For example, consider the following ALLOW macros:

```
ALLOW  2,2,JOBS,JOBNAME,'PAY*',ID='HR*'
ALLOW  3,0,SYSOUTS,CLASS,'P',ID='HR*'
```

The first ALLOW macro permits userids beginning "HR*" to select jobnames beginning "PAY" and browse the log, messages, and JCL data sets. Level 3 display access is required to browse other data sets of the job.

The second ALLOW macro says that any "HR*" user can browse but not modify any sysout data set with sysout class "P" for any job that they are able to select. This SYSOUTS ALLOW macro is required because it grants a higher level of display access to some data sets than the JOBS macro granted.

It is important to point out that granting access with a SYSOUTS ALLOW macro only affects jobs or groups that the user is able to select. Sysout functions are never attempted until after a job or group has been selected.

Using Your Security System to Control IOF Access

You can see that IOF ALLOW macros provide very powerful features for controlling access to IOF resources. However, these macros must be regenerated each time that you need to make a change. This problem can be avoided by using your host security system to control some IOF access

decisions. IOF provides interfaces to RACF, Top Secret, and ACF2 for this purpose.

See Chapter 9 for a description of how to control access to IOF resources with your security system. The discussion which follows is a description of the basic IOF facilities that are involved with providing this support.

You specify which types of resources are to be controlled by your security system by coding ALLOW macros with the ACF parameter. For example, you can request that access to jobs be controlled by your security system but not access to JES2 devices.

For most installations, some reasonable combination of IOF access table control and security system control seems to work best. For example, it usually makes sense to allow users to look at and control their own jobs without checking with the security system. But for looking at other users' jobs, it is often easier to let the security system contain the rules.

To use your security system to control IOF access:

- Define your security system to IOF in options member A60ACF.
- Select the resources to be controlled by modifying options member B24ACFDF.
- Use the ACF command under IOF (while under ISPF) to manage resource names in your security system that correspond to IOF resources that you wish to control.

Each of these topics will be discussed in some detail below.

Defining Your Security System to IOF

To use a security system interface for IOF you must first designate your security system to IOF in the A60ACF options member.

ALLOW Macros to Activate Security System Checks

The ALLOW macros to activate security system checks are automatically generated by choices that you make in options member B24ACFDF. The description below explains how those ALLOW macros relate to your security system.

No security system check will be made unless specifically requested by an ALLOW macro in options member B23ALLOW. Coding the ACF parameter on an ALLOW macro means that the ALLOW macro is actually just a pattern for a rule (profile) that is defined to your security system. The corresponding rule in your security system will be checked to determine if the function requested by the user should be allowed.

For example, consider the following ALLOW macro:

```
ALLOW 3,2,JOBS,JOBNAME,*,ACF=GLOBAL
```

This macro tells IOF to check your security system for rules that control access to jobs based on job name. If a user attempts a job function (like CANCEL or PRINT) and no non-ACF ALLOW macros in B23ALLOW permit the function, a security system check will be made to see if a rule grants the level of access necessary to perform the requested function.

The display and update access levels ("3,2" in our example above) for an ACF ALLOW macro have completely different meanings than for non-ACF ALLOW macros. For ACF ALLOW macros the access levels specify the maximum level of access that can be granted through the security system. Using our example macro above, IOF would never grant more than level 3 display or level 2 update access to a job based on job name rules stored in the security system.

The third parameter of an ACF ALLOW macro ("JOBS" in our example above) must specify a valid IOF resource type (JOBS, GROUPS, DEVICES, SYSOUTS, COMMANDS, or SYSTEMS).

The fourth parameter ("JOBNAME" in our example above) must specify the name of an IOF resource attribute. See the table in the section above, ***IOF Resource Attributes***, for a list of valid resource attributes.

The fifth parameter (the asterisk, *****, in our example above) allows you to restrict security system checks to certain job names. In the example we specified the generic asterisk which means that the security system should be checked for all job names. You could specify instead a list of generic job names and the security system would be checked only if the name of the job being accessed matched one of the generic job names.

ACF=GLOBAL in our example above means that only rules controlled by the system security administrator should be checked. ACF=USER means that rules that can be controlled by individual users should be checked.

The ID, IDLST, GROUP, GRPLST, and other user qualification parms can be specified on an ACF ALLOW macro to limit ACF control to specific groups of users. An ACF ALLOW macro should never be pointed to by the ALLOW or ALOWLST parameters of a GROUP macro.

ALLOW macros with the ACF parameter are checked last, so most common types of access can be granted without a security system call. The default B23ALLOW option contains several ALLOW macros that permit all users to access the jobs they submitted. For efficiency these macros normally should not be removed even when some access control decisions will be made by the security system.

Adding Security System Resource Names

To grant access to an IOF resource you must first add a resource name to your security system that corresponds to the IOF resource. Each ACF

ALLOW macro indicates that there are security system resource names that correspond to the resources described in that ALLOW macro.

For the TSO version of IOF you would normally use the ACF command from any IOF panel (while under ISPF) to add security system resource names. If you have both the TSO and the CICS versions of IOF and your A60ACF and B24ACFDF options members are compatible, you can define your IOF/CICS resource names using IOF/TSO. If you have only the CICS version, you will have to use your security system to manually add resource names. The description below explains the structure of these resource names.

Syntax

```
prefix.table.D/U.resource.attribute.value
```

prefix. The PREFIX= value from options member B24ACFDF. The default value is IOFACF.

table. Specifies the first 3 characters of the access table name from options member B21ACESS that will be used to control this access attempt. The default table name is STANDARD, so this level will normally be "STA".

D/U. "D" means that this resource name will be used to control only display access to the resource. "U" means that this resource name will be used to control only update access to the resource. If a display function (like browse or snap) is being attempted by the user, the resource name that is checked will have "D" in this position. For update functions (like cancel of modify) this level will be "U".

resource. Specifies the first 4 characters of an IOF resource type (JOBS, GROU, DEVI, SYSO, COMM, SYST). This is the type of IOF resource that this security system resource name can be used to control.

attribute. Specifies the first 4 characters of an IOF resource attribute name from the table in "IOF Resource Attributes". Access will be granted based on this attribute (JOBNAME, DEST, etc.).

value. Specifies a specific value for the attribute above. For example, if attribute were "JOBN", this would be a specific (or generic) job name to which permissions are to be granted.

The resource class for IOF resource names is defined by the CLASS= parameter in the A60ACF options member. The default class is DATASET.

The table below describes a few examples of GLOBAL resource names that could be used to grant access to IOF resources.

Resource Name	Access Controlled
IOFACF.STA.D.JOBS.JOBN.PROD*	Display functions for jobs with job name beginning PROD
IOFACF.STA.D.JOBS.JOBC.PAYCL.P*	Display functions for all jobs owned by "PAYCL" with jobnames beginning "P"
IOFACF.STA.U.JOBS.JOBC.PAYCL.*	Update functions for all jobs owned by "PAYCL"
IOFACF.STA.*.JOBS.*	Display and update functions for all jobs in the system
IOFACF.STA.*.GROU.DEST.ATLANTA	Display and update functions for output groups routed to ATLANTA
IOFACF.STA.U.COMM.CMDC.*	Issuing all MVS and JES2 commands from IOF
IOFACF.STA.U.COMM.CMDC.MVS.C.CICS*	Issuing the MVS Cancel command for all jobnames beginning "CICS"
IOFACF.STA.U.COMM.CMDC.JES.*.*	Issuing all JES2 commands and all operands
IOFACF.STA.*.DEVI.DEVN.PRINTER3	Display and update functions for the device named PRINTER3
IOFACF.STA.D.DEVI.*	Display functions for all JES2 devices
IOFACF.STA.*.SYST.SYSI.PROD	Sysplex display and update for system PROD

Global Resource Names Examples

Remember that an ALLOW macro in the B23ALLOW options member is required to activate each of the resource name checks described above.

Now, look at the resource name for output groups above and notice how the resource name reads like a sentence. You would use this resource name to grant "standard display and update functions to all groups with a dest of ATLANTA".

See the table in the section above, ***IOF Resource Attributes***, for a description of all the possible combinations of resources and attributes that can be used in IOF ALLOW macros and security system resource names. In practice, most installations use only a small subset of these combinations. But, it is clear from the table that IOF is extremely flexible in allowing you to define the access rules for your users.

The access control system rule to be checked is completely controlled by the ALLOW macro. For example, assume that the following ALLOW macro is present in B23ALLOW:

```
ALLOW  3,2,JOBS,JOBNAME,'*',ACF=GLOBAL
```

Now, assume that a user attempts to cancel job PAYEDIT. Assume further that no non-ACF ALLOW macros in B23ALLOW permit the user to perform the cancel function, and that no LIMIT macros absolutely prevent the access. IOF will request access from the security system to the following GLOBAL resource name to determine if the user is permitted to cancel PAYEDIT:

```
IOFACF.STA.U.JOBS.JOBN.PAYEDIT
```

The level of access to be checked is discussed in the next section, ***Granting Access to IOF Resources***. If access is allowed, PAYEDIT will be canceled.

You completely control which types of resources are checked by selecting options on the activating ALLOW macros.

If the user had attempted to select job PAYEDIT for display, the resource name checked would be exactly like the one above, with the exception that the "U" level would be "D". We will see below that this allows you four levels of display access and four independent levels of update access to each type of IOF resource.

Granting Access to IOF Resources

For the TSO version of IOF you would normally use the ACF command from any IOF panel (while under ISPF) to grant access to IOF resources. If you have both the TSO and the CICS versions of IOF and your A60ACF and B24ACFDF options members are compatible, you can grant access using IOF/TSO. If you have only the CICS version, you will have to use your security system to manually grant access to IOF resources. The description below provides more detailed information about how access is granted to IOF resource names.

To grant a user access to an IOF resource you permit them access to the security system resource name that corresponds to that IOF resource. The level of IOF access granted is determined by the level of security system access that is granted. For each security system there is a direct correlation between the four levels of IOF access and specific levels of security system access. This correlation is described in the table below.

IOF Level	RACF Access Type	TSS Access Type	ACF Access Type
1	Read	Read	Execute
2	Update	Update	Read
3	Control	Update, Control	Write
4	After	All	Allocate

ACF Access Levels Table

It is important to remember that an IOF access level has no meaning without also indicating whether it is display or update access. For example, the term "level 2 IOF access" has no meaning. You need to say "level 2 display access" or "level 2 update access". This is because there are four independent levels of IOF display and update access, each numbered 1 to 4. For more information, see the table in the section above, ***IOF Access Levels***.

This means that the IOF access levels in the table above do not indicate whether they are for display or update access. From the previous section you will remember that if a user is attempting a display function, a level of ".D." will be included in the security system resource name to be checked. For an update function a level of ".U." will be included.

The presence of the ".D." or ".U." level in the resource name is what controls whether display or update access is being granted. To grant a particular level of display access to a user, you grant him the corresponding security system access level to a resource name with the ".D." level included.

For example, assume that a user attempts to select job PAYEDIT for review and that no non-ACF macros in B23ALLOW permit the access. From the table in the section above, ***IOF Access Levels***, you can see that IOF level 2 display access to job PAYEDIT is required to select it for review. Since a display function is being attempted, the security system resource name to be checked would be:

```
IOFACF.STA.D.JOBS.JOBN.PAYEDIT
```

To check for level 2 access to this resource name, we go to the ***ACF Access Levels Table*** above and find that level 2 IOF access corresponds to RACF update (or ACF2 read) access. So, IOF would check to see if the current user has RACF update (or ACF2 read) access to the resource above. If so, the user would be allowed to select PAYEDIT for review.

Notice that the normal interpretations of the RACF and ACF2 access level names have no meaning at all for IOF. RACF update access is simply used to correspond to IOF level 2 access. The resource name itself actually indicates whether display or update access is being granted. Another example will help to demonstrate this.

Assume that a user attempts to modify the input class of job PAYEDIT. From the table in the section above, ***IOF Access Levels***, you can see that level 3 IOF update access to PAYEDIT is required to modify its input class. Since an update function is being attempted, the security system resource name to be checked would be:

```
IOFACF.STA.U.JOBS.JOBN.PAYEDIT
```

To check for level 3 update to this resource name, we go to ***ACF Access Levels Table*** above and find that level 3 IOF access corresponds to RACF control (or ACF2 write) access. So, IOF would check to see if the current user has RACF control (or ACF2 write) access to the resource. If so, the user would be allowed to change the input class of PAYEDIT.

Security System Access Control Examples

Each example assumes that the user has not been granted access through non-ACF ALLOW macros. The standard B21ACESS option is assumed. It is also assumed that the A60ACF option specifies RACF and the B24ACFDF option specifies a PREFIX=IOFACF and CLASS=DATASET.

Example 1. Let the system security administrator use the security system to control access to all jobs in the system based on job name. The following ALLOW macro will cause the security system to be called when an

attempt is made to access a job by job name and access has not been granted by another ALLOW macro:

```
ALLOW 4,4,JOBS,JOBNAME,*,ACF=GLOBAL
```

Assume that a job with a job name of WEEKLY is selected for review. Job select requires IOF level 2 display access. To get IOF level 2 display access, RACF "update" access is required to the resource name:

```
IOFACF.STA.D.JOBS.JOBN.WEEKLY
```

The user attempting to select WEEKLY for review needs at least "update" RACF access to this resource name or to a generic RACF resource name that includes this name.

If the user wanted to cancel the WEEKLY job, IOF level 2 update access is required. To get IOF level 2 update access, RACF "update" access is required to the resource name:

```
IOFACF.STA.U.JOBS.JOBN.WEEKLY
```

To change the performance group of WEEKLY, IOF level 4 update access is required. Therefore, RACF "alter" access is required to the resource name shown above.

Note how the third level of the resource name changed from "D" to "U" in the examples above to indicate the change from "display" to "update" access. Both examples above required "update" access to the resource name, because IOF level 2 access was needed in both cases.

Now, let's examine the flexibility the ALLOW macro above gives to the security administrator.

- Resource name 'IOFACF.STA.D.JOBS.JOBN.*' can be used to control display access to all jobs.
  - Users permitted RACF "update" access to this resource have level 2 IOF display access to all jobs in the system. They can select any job for review and browse the log, JCL and messages data sets.
  - Users permitted RACF "control" access to this resource have level 3 display access to all jobs. They can browse all data sets of all jobs.
  - No update access to jobs can be granted using this resource name.
- Resource name 'IOFACF.STA.U.JOBS.JOBN.PAY*' can be used to control update access to all jobnames beginning 'PAY'.
  - Users permitted RACF "update" access to this resource have IOF level 2 update access. They can cancel jobs, route them, or release their held data sets. They can also modify several job and data set characteristics. See B21ACESS for a complete description of all functions and parameters allowed with level 2 update access.
  - Users permitted RACF "control" access to this resource have IOF level 3 update access. They can hold jobs, release jobs, restart jobs, and change their input class and priority.

    - Users permitted RACF "alter" access to this resource have IOF level 4 update access. They can set independent mode and performance group.
    - No display access can be granted using this resource name.
- Resource name 'IOFACF.STA.*.JOBS.JOBN.PROD*' can be used to control both display and update access to all jobnames beginning 'PROD'.
    - Users permitted RACF "update" access to this resource have IOF level 2 display and update access. They can browse the log, jcl and messages data sets of jobs, cancel jobs, route jobs, and release their held data sets.
    - Users permitted RACF "alter" access to this resource have IOF level 4 display and update access. They can do anything to 'PROD' jobs.

Example 2. Let all users permit access to their own jobs.

```
ALLOW  3,2,JOBS,JOBNAME,*,ACF=USER
```

This macro lets individual users define security system rules that grant access to their own jobs. Note, however, that users are not permitted to grant display access higher than level 3, nor update access higher than level 2. This restriction prevents end users from permitting themselves to modify priority, class, and performance group of their own jobs.

A user can allow all users to browse the log, JCL, and messages data sets of all his jobs by entering the following RACF command:

```
ADDDSD IOFACF.STA.D.JOBS.JOBN.*   UACC(UPDATE)
```

The user's userid becomes the prefix because the resource name is not enclosed in quotes.

The user can then allow specific users to browse all the sysouts of his jobs by entering the following RACF command:

```
      PERMIT IOFACF.STA.D.JOBS.JOBN.*
ACC(CONTROL) ID(......)
```

The user can also control access to specific jobs he owns by defining and controlling resource names for the specific names of the jobs to be controlled.

Example 3. Let the system security administrator define operators of all devices by device name.

```
ALLOW  4,4,DEVICES,DEVNAME,*,ACF=GLOBAL
```

This ALLOW macro allows the system security administrator to define resource names that can be used to control all devices by device name. Users can be permitted to display all devices and initiators by permitting RACF read (IOF level 1) access to the resource name below. No device control commands or modifications can be permitted by this resource name:

```
IOFACF.STA.D.DEVI.DEVN.*
```

Operators can be allowed to display and update all attributes of all devices and initiators by permitting RACF alter (IOF level 4) access to the resource name below. Note that the third level of the resource name is generic, meaning that both display and update access are being granted.

```
IOFACF.STA.*.DEVI.DEVN.*
```

The remote 35 operator can be permitted to display all attributes of remote 35 devices, to issue most device commands and to alter many device attributes by permitting RACF read (IOF level 1) access to the resource:

```
IOFACF.STA.*.DEVI.DEVN.R35.*
```

The main console operator can be permitted to control all initiators by permitting RACF alter (IOF level 4) access to the resource:

```
IOFACF.STA.*.DEVI.DEVN.INIT*
```

27. Using IOF to Manage a Sysplex Environment

Introduction

IOF provides several functions that allow you to control all your JES2 systems from a single IOF session:

- **MAS** Command. Display basic JES2 information about all systems and easily start or stop any JES2 in the sysplex.
- **AT** Command. Activate an IOF server session on any system in your VTAM network. Access to IOF functions on the server session is carefully controlled. Some of the more useful functions are to:
  - Control JES2 devices defined on another CPU
  - Display CPU and I/O time for jobs running on another CPU
  - Browse sysout data not yet written to spool for jobs running on another CPU

Controlling Access to Sysplex Functions

You have complete control over which users are allowed to use the **MAS** and **AT** commands. By default, all users with TSO operator authority and all started tasks have the authority to use these commands. To change this, remove "SYSTEMS" from the STC= and/or OPER= parms of the SETACF macro in options member A60ACF.

Giving a user access to an IOF server on another machine in the sysplex (with the **AT** command) does not grant the user any privileges on that system. The user will only be allowed to do the IOF functions specifically permitted by the IOF on that system.

The examples below show how to selectively allow specific users to use these commands by adding ALLOW macros to options member B23ALLOW.

Example 1. Allow all users to use the **AT** command for any system in the sysplex, but do not allow them to use the **MAS** command:

```
ALLOW  0,1,SYSTEMS,SYSID,*,ID=*
```

Example 2. Allow selected users to use the **AT** command for the IPO3 system:

```
ALLOW  0,1,SYSTEMS,SYSID,IPO3,ID=(ABC*,XYZ*)
```

Example 3. Absolutely prevent any user except MSTROPER from using the **AT** command or displaying the JES2 system for system IPO2:

```
LIMIT  0,0,SYSTEMS,SYSID,IPO2,XID=MSTROPER
```

Example 4. Allow users with operator authority to use the **MAS** command but not overtype any display fields:

```
ALLOW  4,0,SYSTEMS,SYSID,*,UADS=OPERATOR
```

Example 5. Allow the same access as the macro above, but add permission to use the **AT** command:

```
ALLOW  4,1,SYSTEMS,SYSID,*,UADS=OPERATOR
```

Configuring Your Communications Protocol to Support the AT Command

See Chapter 28 for information about configuring APPC to support the IOF **AT** command. The IOF **AT** command may support additional protocols in the future.

Testing the AT Command

From the *IOF Option Menu*, enter "AT?" to view a menu containing these options:

Option 1. Displays the names and aliases from the B67SERV Option member. If no names are displayed or if changes need to be made, update B67SERV and run an abbreviated IOF generation as described in Chapter 4.

Option 2. Displays the names of systems that have printers and lines attached.

Option 3. **HELP** for the **AT** command. Command syntax and examples are given.

Option 4. **HELP** for the **AT** line command (*Job List Menu* for running jobs, MAS display).

Assuming you are running on system "1" and system "2" has an IOF server defined, enter:

```
AT 2
```

This will start an IOF server session and display the *IOF Option Menu* on system "2". The top left corner of the display will show that the display was built with data from system "2". Note that the first time IOF is initialized on a server from a client IOF session, a server "logon" must be done so a noticeable delay will be seen. The server session is kept active as long as

the client IOF session is active so that subsequent **AT** commands can be executed immediately.

From the server *IOF Option Menu*, you can issue any IOF options for which you are authorized. For example, entering the "PR" option will display the IOF printer panel for printers attached to system "2".

Terminate the remote IOF session exactly like you would terminate any IOF session, by successive **END** commands, **X** on the *IOF Option Menu*, etc. When the remote session terminates, the original IOF panel from which the **AT** command was issued will be redisplayed.

Optionally, you can enter IOF options on the **AT** command. For example, if you enter:

```
AT 2 M
```

the system monitor will be displayed on system "2". See Chapter 21 of the ***IOF User's Guide***,or enter "AT?" for a more complete description of using the **AT** command.

IOF SERVER Command

The IOF **SERVER** command is the base command used for establishing a server connection. The higher level **AT** command uses **SERVER**. When writing REXX execs or clists, it may be necessary to use the base SERVER function. In addition, the **SERVER** command can be used to establish communications with a server that has not been defined in the B67SERV option.

Syntax

```
SERVER  [protocol ADDR(net-address)] /[servname]
        [USER(userid password)]
        [CMD(initcmd)]
        [CLIST/Rexx]
```

---

protocol. The protocol to be used. APPC is currently the only supported protocol.

net-address. The network address. See the SERVER macro in Chapter 28 for a description of APPC network addresses.

servname. A server name or alias defined in the B67SERV option. Either a protocol and net-address, or a servname must be specified, but not both.

userid. The optional userid for the server session. The client's userid will be used if this parm is not specified.

password. The password associated with the userid above.

initcmd. The optional input parms that are passed to the server application when it starts. See the IOF command syntax for a description of the parms that can be specified. Any parm that can be specified on the IOF command can be passed.

CLIST/REXX. Specifies that this **SERVER** command came from a running client clist or REXX exec and that the server session should continue to fetch its input commands from the client clist. **TSICOPY** commands on the server automatically operate against the client clist unless there is also a nested clist started on the server. In this case **TSICOPY** will work by default against the local clist. You can specify TO(CLIENT) on **TSICOPY** commands to set clist variables back on the client from a local clist on the server.

28. Configuring APPC to Support the AT Command

IOF/CICS can operate as a client IOF session but cannot be run as a server. If you have both the CICS and TSO versions of IOF, you may want to review this chapter of the *IOF/TSO Installation Guide*. There, you will find complete information about configuring an IOF/TSO server.