

Steadfast Security in the Software Life Cycle ***CA-Endevor® (SCM for Mainframe) & RACF Unite***

Southern CA Endevor User Group

June 2008



Rose A. Sakach

Endevor Practice Leader - RSH Consulting, Inc.

R.Sakach@RSHConsulting.com - 617-969-9050 - www.rshconsulting.com

Abstract

Steadfast Security in the Software Life Cycle

Software is one of your most valuable assets and therefore deserves the utmost protection from unauthorized changes. Properly implemented, Endeavor and RACF together can provide the necessary protection.

In this session, we will describe an effective and manageable method of configuring both dataset and functional external security with RACF. Topics include implementing the alternate user id, providing appropriate C1DEFLTS table entries, coding the ESI Name Equates Table, choosing the dataset class or a general resource class to define RACF profiles, and creating profiles based on table definitions. We will also cover granting access to software libraries, utilizing the ESI trace facility, and modifying the logging options. Example of tables and RACF profiles will be provided.

AGENDA

A Brief RACF Overview

Dataset Security vs. Functional Security

External Security Interface (ESI)

The Alternate ID

Configuration and Implementation

The Administration Evolution

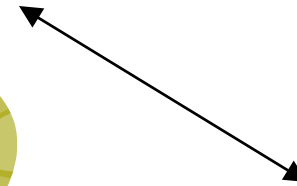
Got Security?

Sample RACF Profiles

A Brief RACF Overview

Resource (Endevor)

RACF



Functions:

- User identification and authentication
- Resource access authorization
- Monitor user activity
- Access administration

A Brief Endeavor Overview

RACF enhances and enforces Endeavor's software controls:

- ✓ **Controls access to managed software items**
- ✓ **Controls use of Endeavor functions**
- ✓ **Limits access to MCF, package dataset, ACMQ and all output libraries to authorized USERIDs only**
- ✓ **Offers RBAC group structure which serves a dual role in providing:**
 - **Access authorization to managed software items**
 - **Authorization to signoff or approve software changes**
- ✓ **Provides the ability to properly manage and administer software access and software change approval**

A Brief RACF Overview

Resource Access Control Facility

IBM's security software product for MVS, OS/390, & z/OS

Component of IBM's Security Server for OS/390 & z/OS

Comprised of:

- **Software - Programs, MACROs, TSO Commands, Utilities**
- **Database (Primary & Backup Pair)**
 - **SETROPTS Options**
 - **Profiles - Users, Groups, Datasets, General Resources**

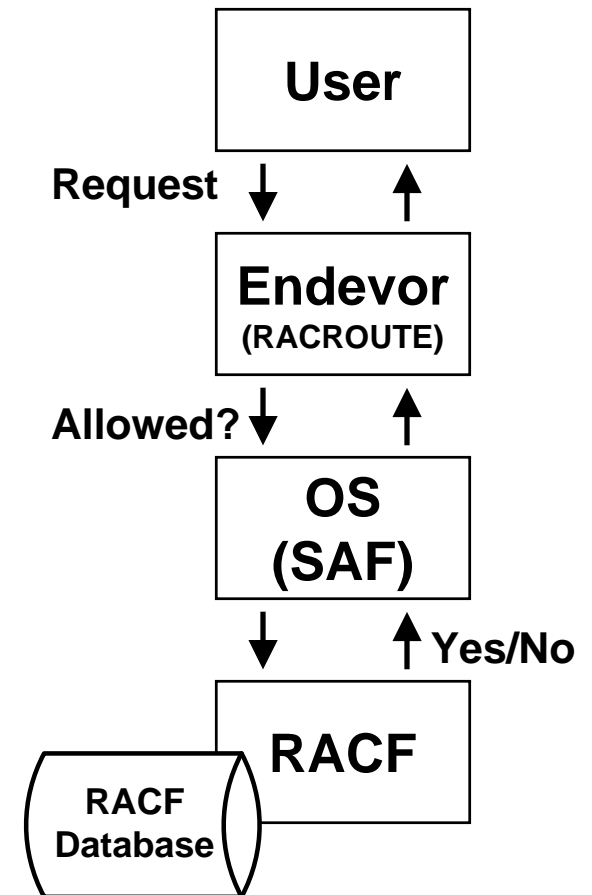
A Brief RACF Overview

RACF is called by a system resource manager (e.g. Endeavor) whenever a user tries to logon or attempts to access a resource

RACF determines whether an action is authorized and *advises* the resource manager to allow or disallow the action

RACF uses the profiles defined in its database to make these determinations

The resource manager *decides* what action to take based on what RACF advises



A Brief RACF Overview

RACF determines whether a user is authorized to access a resource at the requested level of access (e.g., READ) based on resource profiles defined in its database

Resource profiles specify:

- **The logical name of the resource (e.g., the DSNNAME) or mask thereof**
- **The type of resource or 'Class' (e.g., \$ENDEVOR)**
- **How the resource is to be protected and who has access**
- **How access to the resource is to be logged**

Resource profile categories

- **Dataset**
- **General resource (e.g., \$ENDEVOR)**

A Brief RACF Overview

```
RLIST $ENDEVOR C1.*.PMENU.ENVRMENT ALL
```

```
CLASS      NAME
-----
$ENDEVOR   C1.*.PMENU.ENVRMENT
```

```
LEVEL  OWNER      UNIVERSAL ACCESS  YOUR ACCESS  WARNING
-----
  00    END$              NONE              NONE          YES
```

```
INSTALLATION DATA
```

```
-----
NONE
```

```
APPLICATION DATA
```

```
-----
NONE
```

```
SECLEVEL
```

```
-----
NO SECLEVEL
```

```
CATEGORIES
```

```
-----
NO CATEGORIES
```

```
SECLABEL
```

```
-----
NO SECLABEL
```

```
AUDITING
```

```
-----
FAILURES(READ)
```

A Brief RACF Overview

NOTIFY

NO USER TO BE NOTIFIED

CREATION DATE (DAY) (YEAR)	LAST REFERENCE DATE (DAY) (YEAR)	LAST CHANGE DATE (DAY) (YEAR)
270 04	282 04	282 04

ALTER COUNT	CONTROL COUNT	UPDATE COUNT	READ COUNT
000000	000000	000000	000000

USER	ACCESS	ACCESS COUNT
------	--------	--------------

ENDEVOR	READ	
END\$ADM	READ	
JWILLS2	READ	

A Brief RACF Overview

INFORMATION FOR DATASET ENDV.PAY0P.SMS00.LOAD (G)

LEVEL	OWNER	UNIVERSAL ACCESS	WARNING	ERASE
00	ENDV\$	NONE	NO	NO

AUDITING

FAILURES (READ)

NOTIFY

NO USER TO BE NOTIFIED

YOUR ACCESS	CREATION GROUP	DATASET TYPE
READ	\$\$RES	NON-VSAM

GLOBALAUDIT

NONE

NO INSTALLATION DATA

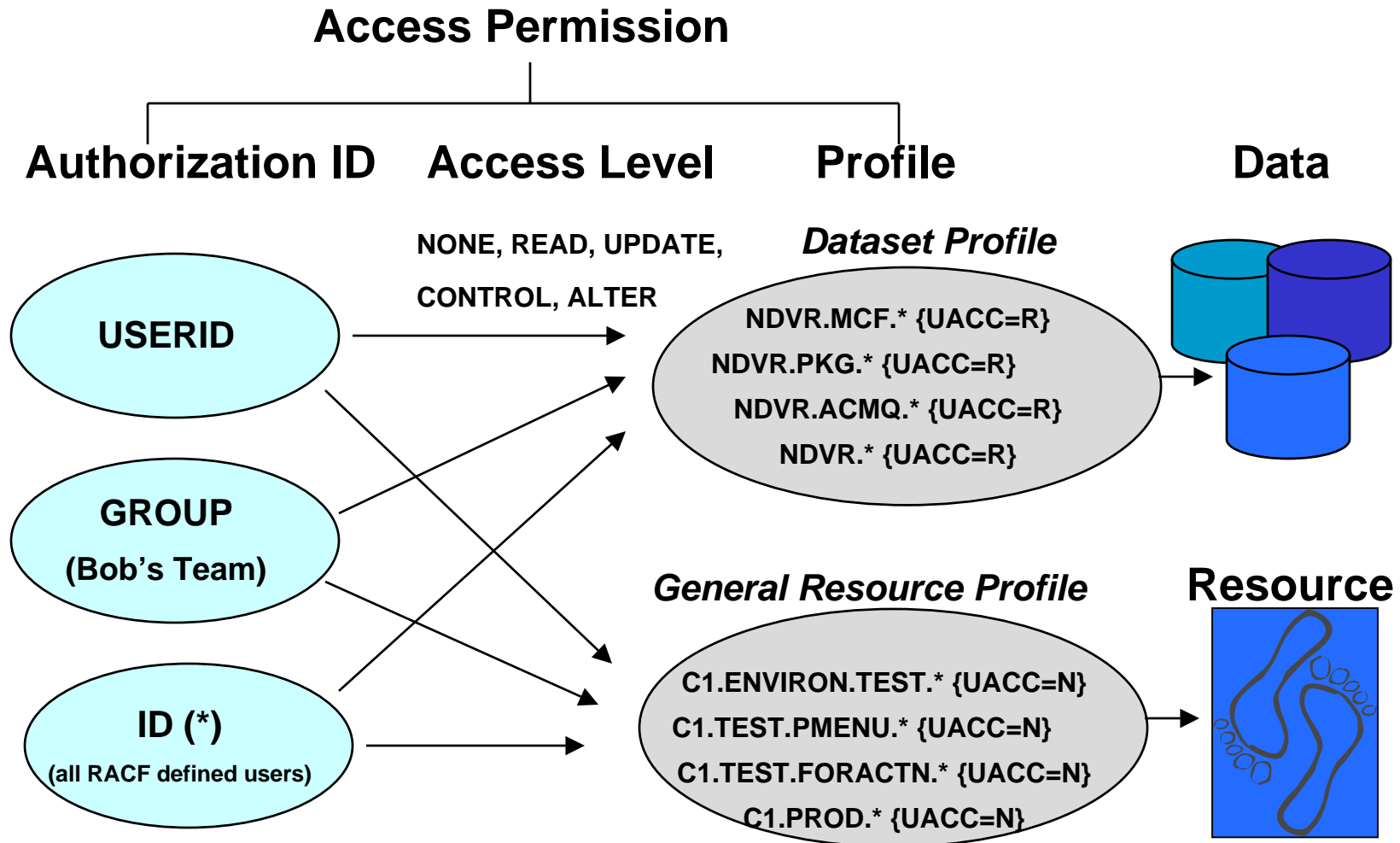
. . .

ID	ACCESS
ENDEVOR	UPDATE
END\$ADM	ALTER
#NOOPER	NONE
ID(*)	READ

ID	ACCESS	CLASS	ENTITY NAME

NO ENTRIES IN CONDITIONAL ACCESS LIST			

A Brief RACF Overview



A Brief RACF Overview

RACF uses the most specific profile (i.e., closest match) for determining access authorization

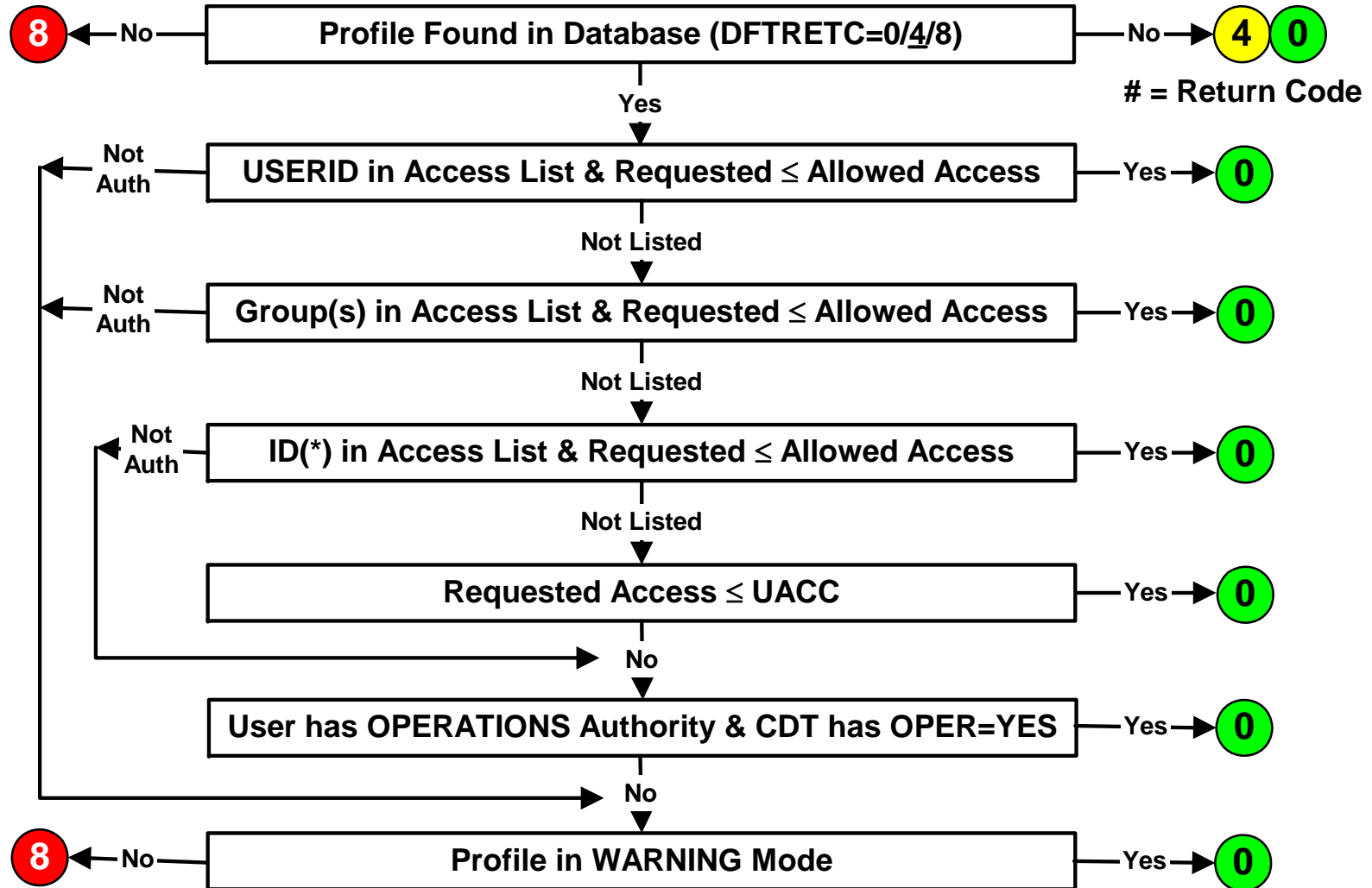
Generic masking characters & order of precedence

- % Single character substitution
- * One full/partial qualifier (general resources - any ending characters)
- ** Zero or more qualifiers

One, and only one, profile protects a given resource - once RACF finds the closest match, it doesn't look any further

ENDV.MCF.TS1.VFILE	←	ENDV.MCF.TS1.VFILE
ENDV.MCF.PS%.**	←	ENDV.MCF.PS1.VFILE
ENDV.MCF.T*.**	←	ENDV.MCF.TS2.VFILE
ENDV.*.P*.**	←	ENDV.PCF.PROD.VFILE
ENDV.**		

A Brief RACF Overview

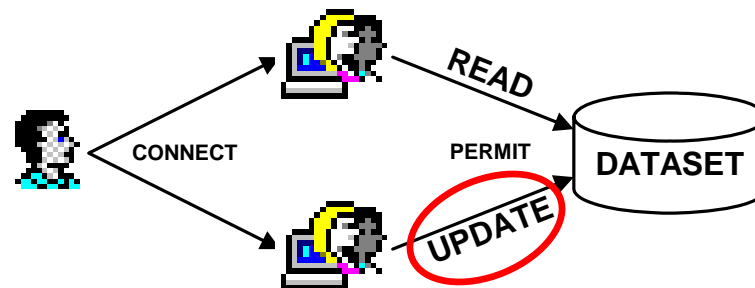


A Brief RACF Overview

Users who are connected to a Group are granted all the access the Group has been permitted

Users may be connected to more than one group with access permissions to the same resource

RACF grants access based on the highest level of access allowed by any of the user's connected groups



RACF & ENDEVOR

RACF merely answers 'yeah' or 'nay' in response to questions (RACROUTE calls) as to whether a given user has sufficient access permission to a particular resource and uses its profiles to determine the answer

Endevor is responsible for posing the question (initiating the RACROUTE call) and uses its configuration table options to formulate the question (RACROUTE operands)

Successful implementation requires the design of harmonious Endevor options and RACF profiles

Dataset vs. Functional Security

Dataset Security

- **Controls all access to Endeavor's data files**
 - **Master Control Files (MCF)**
 - **Package Control Files (PCF)**
 - **Processor Output Libraries**
 - **Base / Delta Libraries**
 - **Output Listing Libraries**
 - **ACMQ Files**
- **Uses RACF DATASET class profiles**
- **Can be implemented regardless of ESI**
- **Should follow data security standards in creating dataset class profiles, establishing groups and granting access:**
 - **Endeavor file naming standards should help ensure minimal RACF profile requirements**
 - **Access should be granted via RACF Groups (not users)**
- **Access authorization can be limited by utilizing the Endeavor Alternate ID**

Dataset vs. Functional Security

Functional Security

- **Controls user access in to Endeavor menus and actions**
 - **Environment Selections**
 - **Primary Options Selections**
 - **Interactive and batch actions**
 - **Package actions**
- **Can use either RACF Dataset or General Resource class profiles**
- **Provides the capability to restrict user access into specific software inventories (Endeavor Systems) ***
- **Provides the mechanism to align Endeavor access with a specific business function**
- **Requires use of the ESI**

***SOX (Sarbanes-Oxley) Requirement**

External Security Interface

What is it?

- An optional feature which provides the capability to integrate and customize Endeavor's functional security with RACF
- Without ESI, aligning Endeavor access to job function would require Native Security table configuration, implementation, maintenance and in some instances, user exits; or may not be possible

ESI Components

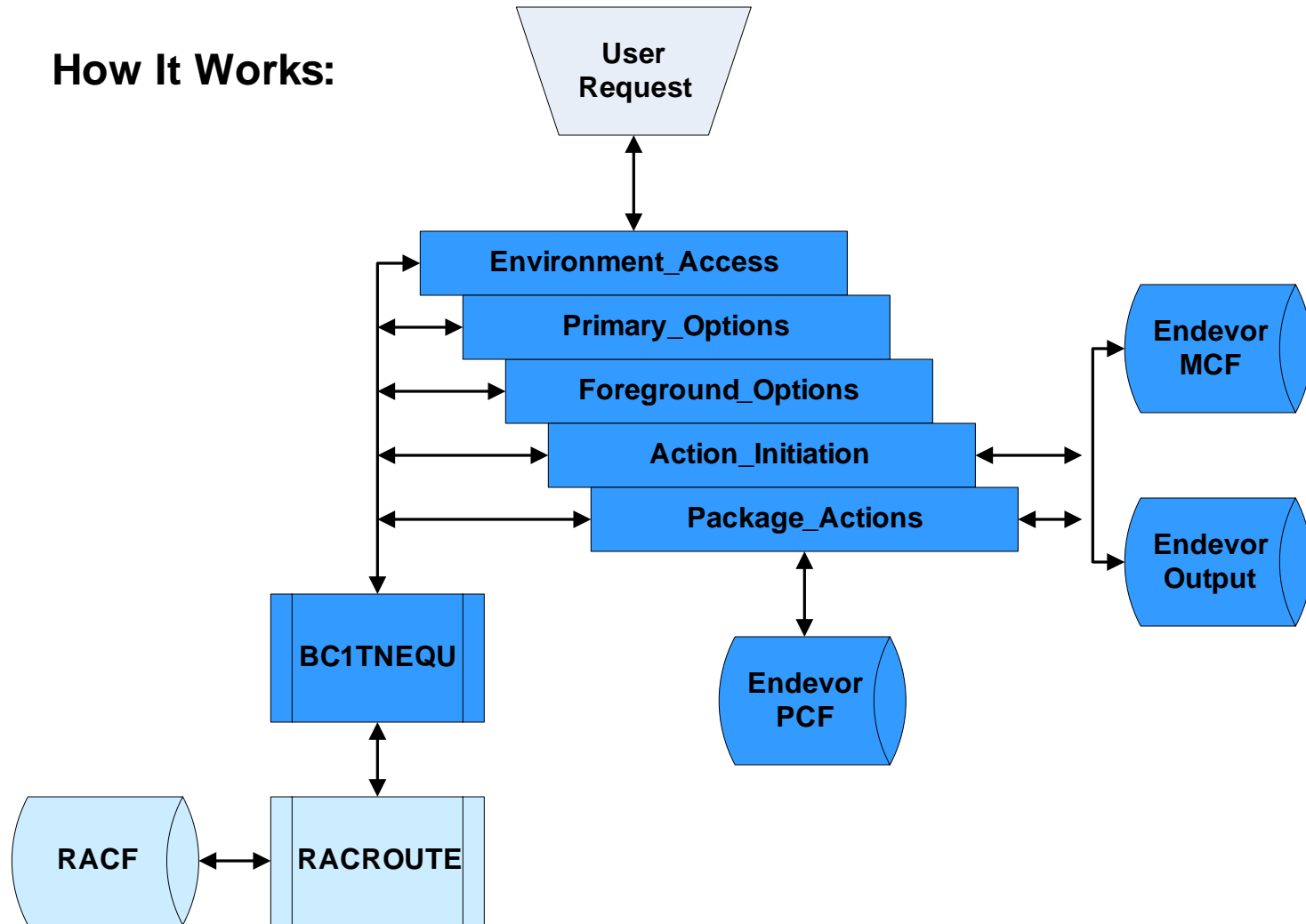
- Default Table (C1DEFLTS)
- Security Table (BC1TNEQU)
 - ESIDFLTS
 - FUNCEQU
 - NAMEQU

Activated via C1DEFLTS parameters

- ESSI=Y
- ACCSTBL=BC1TNEQU

External Security Interface

How It Works:



External Security Interface

Prior to configuration:

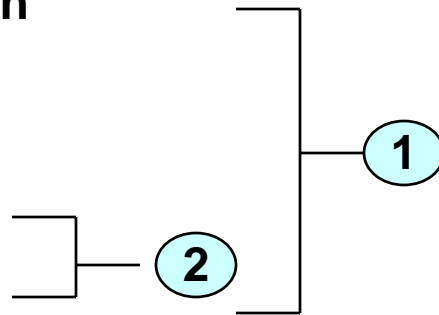
- Understand how the function equates translate to access levels
- Understand how the name equates translate to resource names
- Organize access according to functional requirements (i.e. roles)
- Review Appendix A - *Endevor Change Manager Security Guide*

Business Function	Action Requirement
Endevor Administrator	All
Technical Support	Primary, Limited Foreground, Limited Actions, All Package Options
Operations	Environ (PROD), Limited Primary Options, Limited Package
Application User	Limited Primary, Foreground, Actions, Limited Package
Application Approver	Environ (PROD), Limited Primary, Limited Package
Endevor Sweep Job	All Actions, Package Options

External Security Interface

ESI Security Control Points:

- Environment Selection
- Primary Options
- Foreground Options
- Action Initiation
- Package Actions



- ① **Name Equates Entries:** Construct resource names to represent Endeavor menu options and the actions that can be initiated from them.
- ② **Function Equates Entries:** Assign a RACF access level permission requirement (e.g. READ) to the action initiation and/or package action control points (i.e. DISPLAY, ADD, UPDATE, PCREATE, PREVIEW etc.)

Note: Environment Selection, Primary Options, and Foreground Options control points are automatically assigned a permission requirement of READ.

External Security Interface

A Closer Look at BC1TNEQU:

ESI Defaults

```
TITLE 'BC1TNEQU - EXTERNAL SECURITY INTERFACE TABLE.'  
*****  
* DEFINE ESI DEFAULTS. THE LABEL IDENTIFIES THE TABLE MODULE NAME. *  
*****  
BC1TNEQU ESIDFLT5 WARN=NO, NORMAL EXECUTION MODE X  
                HEADER=YES, PRINT TABLE IN TRACE X  
                LATSIZE=10, 40K (10*4K) LOOK ASIDE TABLE X  
                DESC=6, DESCRIPTOR X  
                ROUTCDE=11, AND ROUTING CODES X  
                TITLE='ENDEVOR 4.0 Security Table'
```

.....
WARN: {YES|NO} Warn mode for entire table - *Note: Individual formats can override*

HEADER: {ALL|YES|NONE} Write header info to trace DSN upon open of EN\$TRESI DD

LATSIZE: {2-10|0|blank} Cache RACF responses – 0 turns off – recommend 10

DESC: {WTO|code} Descriptor code when trace is directed to operator console

ROUTCDE: {WTO|code} Routing code when trace is directed to operator console

External Security Interface

ESI FUNCEQU - Sample

```
*****
*           MAP E/MVS AUTHORITIES TO SAF AUTHORITIES           *
*****
FUNCEQU SAFAUTH=READ,                                         X
                C1ACTNS=( PDISPLAY, PLIST, PREVIEW)
FUNCEQU SAFAUTH=UPDATE,                                       X
                C1ACTNS=( ADD, GENERATE, MOVE, RETRIEVE, UPDATE, SIGNIN,
                DELETE, PCREATE, PMODIFY, PCAST )             X
FUNCEQU SAFAUTH=CONTROL,                                       X
                C1ACTNS=( SIGNOVR )
FUNCEQU SAFAUTH=ALTER,                                         X
                C1ACTNS=( ENVRNMGR, ARCHIVE, TRANSFER,
                PBACKOUT, PCOMMIT, PEXECUTE, PUTILITY )       X
FUNCEQU TYPE=END
```

.....

*** RACF permissions (NONE,READ,UPDATE,CONTROL,ALTER) specified in the FUNCEQU statements determine the minimum level of access authority users require in order to perform specific actions.**

External Security Interface

ESI FUNCEQU

Sets the level of access permission (e.g., UPDATE) required to perform a specific action or package action (e.g., MOVE)

Endevor includes this in the RACROUTE call to RACF

SAFAUTH=auth

- Where 'auth' = NONE | READ | UPDATE | CONTROL | ALTER
- NONE - no security check is issued

C1ACTNS=(c1access,c1access,...)

- Where 'c1access' =

DISPLAY	RETRIEVE	ADD	ENVRNMGR
UPDATE	MOVE	GENERATE	SIGNIN
SIGNOVR	ARCHIVE	DELETE	P...

Each 'c1access' can only be associated with one SAFAUTH

External Security Interface

ESI NAMEQU - Sample

```
NAMEQU ENVIRONMENT_ACCESS, X
    L1=( 'C1' ), X
    L2=( 'ENVIRON' ), X
    L3=( ENVIRONMENT ), X
    LOG=NOFAIL, X
    WARN=NO, X
    CLASS= ' $ENDEVOR '
NAMEQU PRIMARY_OPTIONS, X
    L1=( 'C1' ), X
    L2=( ENVIRONMENT ), X
    L3=( 'PMENU' ) X
    L4=( MENUITEM ), X
    LOG=NOFAIL, X
    WARN=NO, X
    CLASS= ' $ENDEVOR '
.
.
```

External Security Interface

ESI NAMEQU – Format Statement Syntax

Defines the construct of resource names for RACF calls

FORMATn | name

- **FORMAT** statement number, where n = 1-5 (Pre-3.9 structure)
- **FORMAT** name, where name =
 - **ENVIRONMENT_ACCESS**
 - **PRIMARY_OPTIONS**
 - **BACKGROUND_OPTIONS**
 - **ACTION_INITIATION**
 - **PACKAGE_ACTIONS**

Ln=(field1(begin,length),...fieldn(begin,length))

- **Index** level number, where n=1-10
- **'Fieldn'** is a literal (e.g., 'PMENU') or keyword variable (e.g., SYSTEM)
- **'Begin,Length'** optionally specifies a keyword substring

External Security Interface

KEYWORD VARIABLE	NAMEQU STATEMENT WHERE APPLICABLE				
	ENV	PRI	FOR	ACTS	PKG
ENVIRONMENT	X	X	X	X	
ACTION		X	X	X	X
MENUITEM		X	X	X	X
MENUAUTH		X	X	X	X
SYSTEM				X	
SUBSYSTEM				X	
STAGEID				X	
STAGENO				X	
STAGENAME				X	
TYPE				X	
ELEMENT				X	
ELM-10				X	
CCID				X	
PKGSUBFC					X
PKGID					X
PKGTYPE					X
PKGSTAT					X
PKGAPPGR					X
PKGBOE					X

External Security Interface

ESI NAMEQU – Format Statement Syntax (more...)

CLASS=DATASET | classname (e.g., \$ENDEVOR)

LOG=NONE | ASIS | NOFAIL | NOSTAT

- Determines LOG= keyword on RACROUTE Macro
- Recommend NOFAIL for menu-related and ASIS for all others

WARN=YES | NO

- Overrides ESIDFLTS option
- If not coded, ESIDFLTS option is used

External Security Interface

ESI NAMEQU - RACF Resource Class Options

DATASET Class {default}

- **Limits length of profile to 44 chars (dataset name limitation)**
- **Updates involve communication and coordination of class refresh**
- **Can cause an increase in RACF I/O:**
 - **Can't be RACLISTED (each user must fetch set of profiles)**
 - **REFRESH forces every user to drop & rebuild all DATASET profiles**

General Resource Class {i.e. \$ENDEVOR}

- **Permits 246 char profile length (more descriptive names)**
- **Can improve performance:**
 - **Able to utilize discrete profiles more easily**
 - **Able to be RACLISTED if class is properly configured**
 - **REFRESH is less disruptive (eliminates logoff/logon user requirement)**
- **Offers better stability and affords faster change process**

External Security Interface

ESI NAMEQU - ENVIRONMENT_ACCESS

```
*****  
*           SPECIFY SAF DATASET NAME FORMATS           *  
*****  
  
NAMEQU ENVIRONMENT_ACCESS,                                X  
    L1=( 'C1' ),                                          X  
    L2=( 'ENVIRON' ),                                     X  
    L3=( ENVIRONMENT ),                                   X  
    LOG=NOFAIL,                                          X  
    WARN=NO,                                             X  
    CLASS= ' $ENDEVOR '
```

Translates to: **C1.ENVIRON.environment**
Where: 'environment' = TEST, PROD, etc.
Example: **C1.ENVIRON.TEST**
Profile: **C1.ENVIRON.*** (* - PROD,TEST)

External Security Interface

ESI NAMEQU - PRIMARY_OPTIONS

```
NAMEQU PRIMARY_OPTIONS, X
      L1=( 'C1' ), X
      L2=( ENVIRONMENT ), X
      L3=( 'PMENU' ), X
      L4=( MENUITEM ), X
      LOG=NOFAIL, X
      CLASS= '$ENDEVOR`
```

.....

Translates to: C1.environment.PMENU.menuitem

Where: 'environment' = TEST, PROD, etc
'menuitem' = DISPLAY, FOREGRND, BATCH,
PACKAGE, BATCHPKG, USER,
ENVRMENT, UNLOAD, RELOAD

Example: C1.TEST.PMENU.DISPLAY

External Security Interface - NAMEQU

ESI NAMEQU - PRIMARY_OPTIONS

SAMPLE RESOURCES

C1.PROD.PMENU.BATCH
C1.PROD.PMENU.BATCHPKG
C1.PROD.PMENU.DISPLAY
C1.PROD.PMENU.ENVRMENT
C1.PROD.PMENU.FOREGRND
C1.PROD.PMENU.PACKAGE
C1.PROD.PMENU.USER
C1.TEST.PMENU.BATCH
C1.TEST.PMENU.BATCHPKG
...
C1.TEST.PMENU.USER

SAMPLE PROFILES

C1.PROD.PMENU.PACKAGE
C1.*.PMENU.PACKAGE
C1.*.PMENU.USER
C1.*.PMENU.ENVRMENT
C1.*.PMENU.BATCHPKG
C1.*.PMENU.*

External Security Interface

ESI NAMEQU - FOREGROUND_OPTIONS

```
NAMEQU FOREGROUND_OPTIONS, X
      L1=( 'C1' ), X
      L2=( ENVIRONMENT ), X
      L3=( 'FORACTN' ), X
      L4=( MENUITEM ), X
      LOG=NOFAIL, X
      CLASS= '$ENDEVOR'
```

.....

Translates to: C1.environment.FORACTN.menuitem

Where:

'environment' = TEST, PROD, etc.

**'menuitem' = DISPLAY, ADDUPDT, RETRIEVE,
GENERATE, MOVE, DELETE,
PRINT, SIGNIN**

Example: C1.TEST.FORACTN.RETRIEVE

External Security Interface - NAMEQU

ESI NAMEQU - FOREGROUND_OPTIONS

SAMPLE RESOURCES

C1.PROD.FORACTN.ADDUPDT
C1.PROD.FORACTN.DELETE
C1.PROD.FORACTN.DISPLAY
C1.PROD.FORACTN.GENERATE
C1.PROD.FORACTN.MOVE
C1.PROD.FORACTN.PRINT
C1.PROD.FORACTN.RETRIEVE
C1.PROD.FORACTN.SIGNIN
C1.TEST.FORACTN.ADDUPDT
C1.TEST.FORACTN.DELETE
...
C1.TEST.FORACTN.SIGNIN

SAMPLE PROFILES

C1.PROD.FORACTN.DISPLAY
C1.PROD.FORACTN.PRINT
C1.PROD.FORACTN.SIGNIN
C1.PROD.FORACTN.*
C1.TEST.FORACTN.*
C1.*.FORACTN.*

External Security Interface

ESI NAMEQU - ACTION_INITIATION

```
NAMEQU ACTION_INITIATION, X
      L1=( 'C1' ), X
      L2=( ENVIRONMENT ), X
      L3=( SYSTEM ), X
      L4=( SUBSYSTEM ), X
      L5=( MENUAUTH ), X
      LOG=ASIS, X
      CLASS= '$ENDEVOR'
```

Translates to: C1.environment.system.subsystem.menuauth

Where: 'environment' = TEST, PROD, etc

'system' = FINANCE, ...

'subsystem' = ACCTPAY, ...

'menuauth' = ADD, ARCHIVE, DELETE, ...

Example: C1.TEST.FINANCE.ACCTPAY.DELETE

External Security Interface

ESI NAMEQU - ACTION_INITIATION

SAMPLE RESOURCES

C1.PROD.FINANCE.ACCTPAY.ADD
C1.PROD.FINANCE.ACCTPAY.ARCHIVE
C1.PROD.FINANCE.ACCTPAY.DELETE
C1.PROD.FINANCE.ACCTPAY.DISPLAY
C1.PROD.FINANCE.ACCTPAY.ENVRNMGR
C1.PROD.FINANCE.ACCTPAY.GENERATE
C1.PROD.FINANCE.ACCTPAY.MOVE
C1.PROD.FINANCE.ACCTPAY.RETRIEVE
C1.PROD.FINANCE.ACCTPAY.SIGNIN
C1.PROD.FINANCE.ACCTPAY.SIGNOVR
C1.PROD.FINANCE.ACCTPAY.UPDATE
C1.TEST.FINANCE.ACCTPAY.ADD
C1.TEST.FINANCE.ACCTPAY.ARCHIVE
...
C1.TEST.FINANCE.ACCTPAY.UPDATE

SAMPLE PROFILES

C1.PROD.FINANCE.ACCTPAY.DELETE
C1.*.FINANCE.ACCTPAY.*
C1.*.FINANCE.*.*

C1.*.*.*.ENVRNMGR
C1.**

External Security Interface

ESI NAMEQU - PACKAGE_ACTIONS

```
NAMEQU PACKAGE_ACTIONS,                                     X
    WARN=YES,                                             X
    L1=( 'C1' ),                                         X
    L2=( 'PACKAGE' ),                                    X
    L3=( MENUAUTH ),                                     X
    LOG=ASIS,                                             X
    CLASS= '$ENDEVOR'
NAMEQU TYPE=END
```

Translates to: C1.PACKAGE.menuauth

Where: 'menuauth' = PCREATE, PCAST, PDISPLAY, ...

Example: C1.PACKAGE.PUTILITY

External Security Interface - NAMEQU

ESI NAMEQU - PACKAGE_ACTIONS

Requires additional activation in C1DEFLT5 table:

- **PKGSEC=ESI or PKGSEC=MIGRATE**

Take advantage of FUNCEQU and control access authorization via 1 profile instead of 10 using varying access levels

Example Package_Actions Profile = **C1.PACKAGE.***

READ	UPDATE	ALTER
C1.PACKAGE.pdisplay	C1.PACKAGE.pcreate	C1.PACKAGE.pcommit
C1.PACKAGE.plist	C1.PACKAGE.pmodify	C1.PACKAGE.pbackout
C1.PACKAGE.preview	C1.PACKAGE.pcast	C1.PACKAGE.pexecute
		C1.PACKAGE.putility

The Alternate ID

Feature that can be used to restrict update access to Endeavor managed software libraries

An Alternate ID, specified in the customer default table, is used to access Endeavor libraries instead of individual users' IDs

Can require updates to controlled datasets be performed only through Endeavor and not outside of Endeavor, thereby ensuring Endeavor file integrity

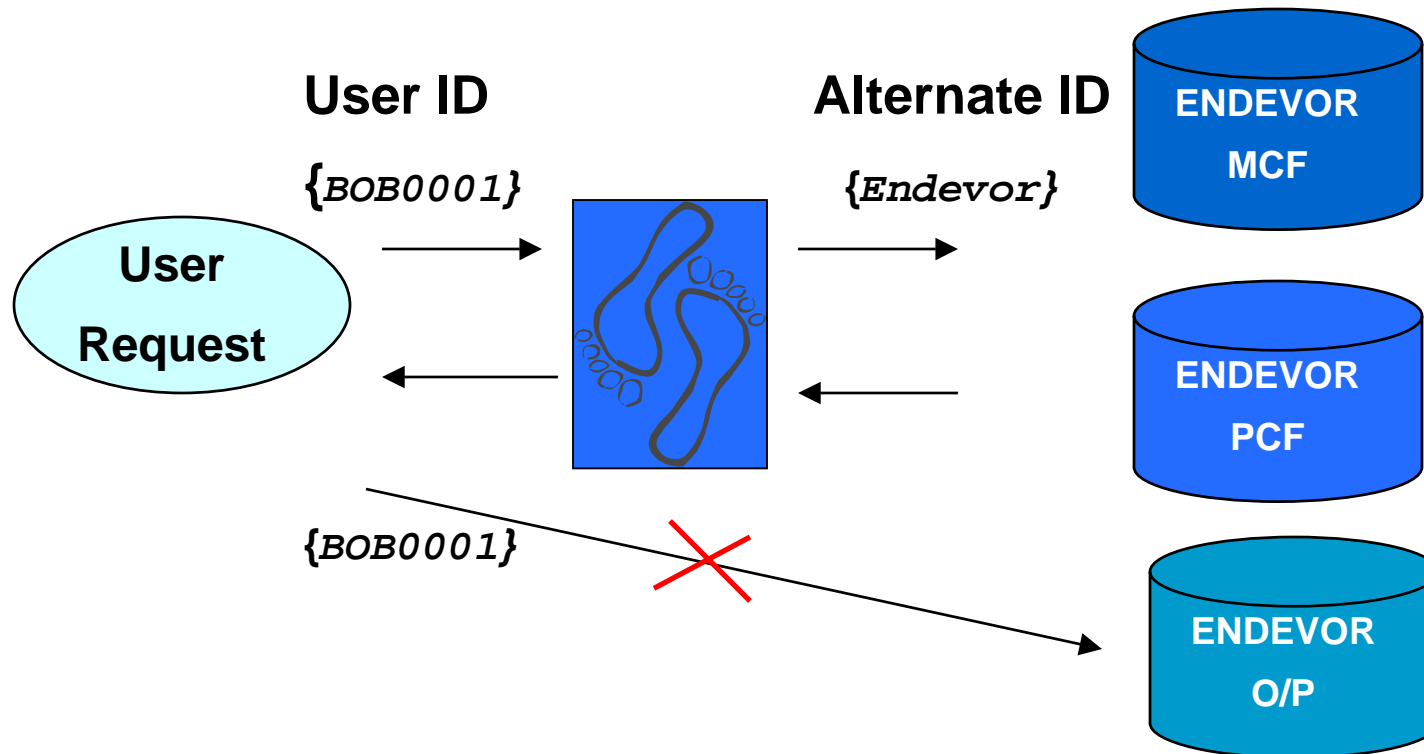
Eliminates requirement to maintain Endeavor dataset access for all users

Utilized for dataset security (and functional security if used as the “sweep job” ID)

Recommended by CA

The Alternate ID

How it works:



Direct access attempts will always utilize the user's personal ID

The Alternate ID

Parameters in the C1DEFLT5 table

- **RACFUID** Userid used for dataset authorization checking
- **RACFPWD** Password for the RACFUID (optional)
- **RACFGRP** RACF group associated with the RACFUID (optional)

Example of the C1DEFLT5 TABLE

```
C1DEFLT5      TYPE=MAIN  
              ESSI=Y  
              RACFUID=userid  
              RACFPWD=clear-text password  
              RACFGRP=logon groupid
```

Datasets containing C1DEFLT5 TABLE source code and assembly module should be protected with a UACC of NONE and strictly limited access if a password is coded

The Alternate ID

A password used to be required but recent changes to Endeavor made it unnecessary - do not use / remove if currently in use

Coding a group unnecessarily locks the ID to a particular group

In RACF

- **Make the Alternate ID PROTECTED (no password logon allowed) assuming no RACFPWD is coded in C1DEFLTS**
- **Grant it access to Endeavor datasets**

If PKGSEC = APPROVER, include it in all Approver groups

The Alternate ID

Endevor switches to the Alternate ID to access the following:

- **Master Control File**
- **Package Control File**
- **ACMQ root and XREF data sets (Note: R4.0 - R7.4 requires ICHRCX01)**
- **Base/Delta libraries**
- **Source Input libraries (e.g. include libraries)**
- **Source output libraries (e.g. load libraries)**
- **All processor datasets (not created or deleted)**
- **CCID validation dataset**
- **Package ship staging data sets**

The Alternate ID requires appropriate access to these datasets

The USERID performing the action requires no access



Note: No Alternate ID swap occurs when ELIB datasets are controlled by CA-L-Serv

The Alternate ID

The Alternate ID is not used to access the following:

- Foreground ISPF work data sets (e.g. uid.C1TEMPRnMSGs, uid.C1TEMPncntl, uid.C1#NTMPL.LIST, etc.)
- ADD/UPDATE FROM: data set
- RETRIEVE TO: data set
- ARCHIVE/TRANSFER TO: data set
- COPY FROM: data set
- RESTORE FROM: data set
- LIST FROM: data set
- TRANSFER FROM: data set
- PRINT TO: data set
- BATCH request data sets

The USERID performing the action requires appropriate access to these datasets

Configuration and Implementation

Design Endeavor BC1TNEQU entries and corresponding RACF profiles

Add General Resource class to RACF if necessary

Create RACF profiles

- **Define Alternate ID**
- **Define groups**
- **Add members to groups**
- **Define resource profiles**
- **Grant access permissions**

Update Endeavor BC1TNEQU

Update C1DEFLT

Configuration and Implementation

On Testing

- **Utilize WARN mode - this ensures all access is allowed and will not disrupt normal activity**
 - **BC1TNEQU ESIDFLTS WARN=YES**
 - **ALD profile-name WARNING**
- **Utilize the Endeavor TRACE facility for ESI:**
 - **To review entity names (i.e. pseudo data sets) and the SAF return codes at each security control point**
 - **Ensure the pseudo data set name is being constructed and evaluated as intended**
- **Monitor security reports and adjust access accordingly**
 - **ENRASW00 = Endeavor ESI Exception Warning report**
 - **RACFRW = RACF Report Summary of all warnings by resource name (use SELECT PROCESS WARNINGS; EVENT ALLSVC)**



Remember! RACF resource profile modifications may require the user to logoff/logon TSO and/or a SETROPTS REFRESH in order to take effect

Configuration and Implementation

The Endeavor Trace Facility

- Don't be afraid to use it!
- Getting Started (see Using the Trace facility in the Endeavor Security Guide manual – ENSECnnn for details)
 - Activate ESI
 - Ensure you have access to Endeavor CLIST library (or utilize the TSO allocate command)
 - Determine foreground or batch mode (determined by how you will be submitting Endeavor actions)
 - Foreground
 - ◆ %ESITRACE START
 - Writes trace records to your terminal
 - ◆ %ESITRACE DATASET DSN(ESI.TRACE) NEW CAT
 - Writes trace records to dsn = userid.ESI.TRACE
 - ◆ %ESITRACE STOP

Configuration and Implementation

The Endeavor Trace Facility (more...)

- **Batch**
 - ◆ **Pre-allocate your trace file; DCB=(RECFM=FBA, LRECL=133, BLKSIZE=1330 {or any multiple of 133})**
 - ◆ **Complete JOB STATEMENT INFORMATION on the BATCH OPTIONS MENU panel**
 - ◆ **Select option #5 (BUILD JCL – Enter additional JCL to be included with the job) from the Batch Options Menu panel**
 - ◆ **Include one of the following JCL statements:**
 - **//EN\$TRESI DD SYSOUT=***
 - **//EN\$TRESI DD DSN=userid.ESI.TRACE, DISP=SHR**

Configuration and Implementation

The Endeavor Trace Facility (more...)

- **Sample output**

- The first “n” ENCS001I: statements will include a listing of the BC1TNEQU table – including the TITLE, the date/time of the last assembly, the ESIDFLTS values, the FUNCEQU entries, and the NAMEQU entries
- The ENCS101I statements follow, and these statements contain the trace information – beginning with the format number:

```
ENCS101I Format=0001 Pass=0000 Auth=READ ACEE=00000000
ENCS101I Class=$ENDEVOR Log=NONE
ENCS101I Scale=0.....1.....2.....3.....4.....+...
ENCS101I Entity=C1.ENVIRON.PROD
ENCS101I User LANCE00 access is denied from SAF
ENCS101I RACROUT RC=0008 RACHECK RC=0008 REASON=0000
```

Configuration and Implementation

The Endeavor Trace Facility (more...)

- Sample output (more...)

- What does it mean?

```
ENCS101I Format=0002 Pass=0000 Auth=READ ACEE=00000000
ENCS101I Class=$ENDEVOR Log=NOFAIL
ENCS101I Scale=0.....1.....2.....3.....4.....
ENCS101I Entity=C1.TEST.PMENU.DISPLAY
ENCS101I User LANCE00 access is allowed from SAF
.....
```

NAMEQU format # where:

- 1=Environment_Access
- 2=Primary_Options
- 3=Foreground_Options
- 4=Action_Initiation
- 5=Action_Initiation
- 6=Package_Actions

Always 0 or 1

Required authority as defined in FUNCEQU

RACF Resource Class

Pseudo DSN passed to SAF

Always = 0 for TSO

SAF or LAT

The Administration Evolution

**“An achievement, no matter how magnificent, will eventually decay if not preserved with proper care”
-George Sheehan**

“Proper security administration does not happen overnight. It is a dynamic process which continuously evolves and must never be marked complete.” -RSH Consulting

The Administration Evolution

Guidelines to ensure security controls function efficiently, effectively and remain in tact:

- **Produce and monitor security reports regularly (statistical reports always gain management attention)**
- **Perform annual Endeavor security reviews**
 - **Ensure functional security rules continue to map to real functions (reorganizations can eliminate functional areas)**
 - **Ensure appropriate user ids are permitted access to groups while inappropriate user ids are removed**
 - **Modify the security table when necessary**
- **De-centralize security administration**
 - **Data Owners determine access and/or approval authorization**
 - **Team Leads grant access and maintain access lists via connect authority**

Got Security?

Signs of proper controls and security:

- ✓ Access to Endeavor's control libraries is restricted to the IDs of the users responsible for maintaining them
- ✓ Access to Endeavor's output libraries is restricted to the Alternate ID and Endeavor administrators
- ✓ Access to Endeavor panels, managed software items, and menu actions is granted using role-based security groups corresponding to user job functions
- ✓ Joining and removing users on application teams is handled simply by making changes in group membership
- ✓ Group changes to join and remove users on application teams automatically updates Approvers as well
- ✓ The business area responsible for the application software is the "stopgate" that decides who can view, modify, or delete code and approve packages; they also have an oversight role to ensure access limitations are enforced

Sample RACF Profiles

External Security Interface

Sample RACF group definitions

```
AG END$          DATA `Endevor Owner'          OWNER(SYS1) SUPGROUP(SYS1)
AG END$ADMN DATA `Endevor Administrator'      OWNER( END$ ) SUPGROUP( END$ )
AG END$TECH DATA `Endevor Tech Support'      OWNER( END$ ) SUPGROUP( END$ )
AG END$USER DATA `Endevor Menus-All users'    OWNER( END$ ) SUPGROUP( END$ )
AG END$APRV DATA `Endevor Pkg Approvers'      OWNER( END$ ) SUPGROUP( END$ )
AG END$TRNG DATA `Endevor Student'           OWNER( END$ ) SUPGROUP( END$ )
AG END$AAA1 DATA `Endevor AAA Level 1'        OWNER( END$ ) SUPGROUP( END$ )
AG END$ESI  DATA `Endevor Alternate Id'       OWNER( END$ ) SUPGROUP( END$ )
```

.....

Notes:

- Group entries should include a description in the data field, an owner, and a superior group entry
- Names and group structure are examples only - following local naming conventions

Sample RACF Profiles

External Security Interface

Sample RACF general resource profile definitions

```
RDEF $ENDEVOR `C1.ENVIRON.**'      OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.ENVIRON.PROD**'  OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.ENVIRON.TEST**'  OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.ENVIRON.TRAIN**'  OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.**'              OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.*.PMENU.**'      OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.*.FORACTN.**'     OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.*.*.**'          OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.*.aaa.**'        OWNER( END$ ) UACC( NONE ) GENERIC
RDEF $ENDEVOR `C1.PACKAGE.**'       OWNER( END$ ) UACC( NONE ) GENERIC
```

Notes:

- "Catch all" profiles - like C1.** - ensure all resources are covered
- Default access UACC specification for ALL users
- Security profiles which include the Endeavor SYSTEM id provide limited inventory access
- Security specifications for package actions limit users ability to perform package functions

Sample RACF Profiles

External Security Interface

Sample RACF access permissions

```
PE `C1.ENVIRON.**' CLASS($ENDEVOR) ACCESS(READ) ID(*)or  
ID(END$ADMN)  
ID(END$USER)
```

```
PE `C1.ENVIRON.PROD**' CLASS($ENDEVOR) ACCESS(READ) ID(END$USER)  
ID(END$ADMN)
```

```
PE `C1.ENVIRON.TEST**' CLASS($ENDEVOR) ACCESS(READ) ID(END$USER)  
ID(END$ADMN)
```

```
PE `C1.ENVIRON.TRAIN**' CLASS($ENDEVOR) ACCESS(READ) ID(END$TRNG)  
ID(END$ADMN)
```

.....
Notes:

- Determine access requirements – should all users have access to all Endeavor menus?
- ID (*) grants access to all users defined to RACF
- ID(END\$USER) limits access to ENDEVOR users

Sample RACF Profiles

External Security Interface

Sample RACF access permissions (more...)

```
PE 'C1.*.PMENU.**' CLASS($ENDEVOR) ACCESS(READ) ID(*) or ID(END$USER)
```

```
PE 'C1.*.FORACTN.**' CLASS($ENDEVOR) ACCESS(READ) ID(*) or ID(END$USER)
```

```
PE 'C1.*.aaa.**' CLASS($ENDEVOR) ACCESS(READ) ID(END$AAA1)
```

```
ACCESS(UPDATE) ID(END$AAA2)
```

```
ACCESS(CONTROL) ID(END$AAA3)
```

```
ACCESS(ALTER) ID(END$ADMN)
```

```
ACCESS(ALTER) ID(END$ESI)
```

```
PE 'C1.*.*.**' CLASS($ENDEVOR) ACCESS(ALTER) ID(END$ADMN)
```

```
ACCESS(ALTER) ID(END$ESI)
```

.....

Notes:

- Application team access will vary depending upon job function
 - Level 1 = display/view element inventory and package approver
 - Level 2 = all functions required to modify elements within the inventory
 - Level 3 = override signout

Sample RACF Profiles

External Security Interface

Sample RACF access permissions (more...)

```
PE 'C1.PACKAGE.**' CLASS($ENDEVOR) ACCESS(READ) ID(END$AAA1)
ID(END$APRV)
```

```
PE 'C1.PACKAGE.**' CLASS($ENDEVOR) ACCESS(UPDATE) ID(END$AAA2)
```

```
PE 'C1.PACKAGE.**' CLASS($ENDEVOR) ACCESS(ALTER) ID(END$ADMN)
ID(END$ESI)
```

.....

Notes:

- Package function abilities are determined by access level (see FUNCEQU)
 - READ: Package Display, Package List, Package Review
 - UPDATE: Create, Modify, Cast
 - ALTER: Backout, Commit, Execute, Utility
- PKGSEC = ESI must be specified in C1DEFLT5 in order to utilize all ESI package action rules. Package actions are usually restricted to users specified in the approver group.

Sample RACF Profiles

External Security Interface

Sample RACF user-to-group connects

```
CO ROSE001 GROUP(EN$ADMN) AUTH(CONNECT) OWNER(EN$ADMN)
CO BOB0001 GROUP(EN$AAA1) AUTH(CONNECT) OWNER(EN$AAA1)
CO MARK001 GROUP(EN$AAA2) AUTH(USE) OWNER(EN$AAA2)
CO BOB0001 GROUP(EN$AAA2) AUTH(CONNECT) OWNER(EN$AAA2)
CO BOB0001 GROUP(EN$AAA3) AUTH(CONNECT) OWNER(EN$AAA3)
CO BILL001 GROUP(EN$AAA3) AUTH(USE) OWNER(EN$AAA3)
```

.....

Notes:

- AUTH(CONNECT) enables the user to connect other users to the group
- Team Leads and/or data owners should be granted connect authority
- EN\$AAA1 Group can serve as the approver group for the application team
- Application specific function abilities are determined by access level (see FUNCEQU)
 - READ (level 1): View / Display
 - UPDATE (level 2): Add, Generate, Move, Retrieve, Update, Signin, Delete
 - CONTROL (level 3): Override Signout
 - ALTER: Envrmgr,Transfer,Archive,Restore

Questions & Answers