CA IDMS Security

Laura Rochon
Hera Evolution LLC

IUA/CA IDMS™ Technical Conference May 7-11, 2018

Abstract

- This class will focus on the methodology, the tools and techniques used to secure access to the CA IDMS runtime system, the database and functional tools. More specifically, we will talk about the different levels and types of security within CA IDMS, how to activate CA IDMS security, how to secure the different types of resources, how the signon process works, how to secure dictionaries and how to report on security.

Laura Rochon

- Laura has worked with CA IDMS for over 30 years, including close to 7 years with Cullinet and CA. Laura is a frequent presenter at CA World and User Conferences in both North America and Europe. As a system and application DBA, Laura has supported multiple clients in North America, by teaching classes, performing database and system reviews, installation and maintenance, and just normal DBA work. She presently works for Hera Evolution LLC, a leader in CA IDMS Support.

Agenda

1. INTRODUCTION
2. SRTT TABLE
3. INTERNAL SECURITY
4. EXTERNAL SECURITY
5. SIGNON PROCESS
6. MINIMUM SECURITY RECOMMENDATIONS
7. APPLICATION SECURITY

Introduction

- CA IDMS Central Security System
- Terminology
- Architecture

CA IDMS Central Security System

Why Secure your system ?

- Protect confidential information
- Maintain integrity of your corporate databases
- Prohibit or deter unauthorized access
- Meet security standards
- Fulfill goverment requirements
- Adhere to privacy Laws

CA IDMS Central Security System (cont'd)

Security Strategy?

- Physical access to computer room
- Electronic access to computer room
- Access to hardware
- Access to software
- Access to databases
- Access to applications
- Access to data sets
- Access to production, QA, and test systems

CA IDMS Central Security System (cont'd)

CA IDMS Centralized Security Administration :

- Can interface with an external security software system to protect CA IDMS resources
- Can protect CA IDMS resources when an external security system is not available or not used to protect CA IDMS resources
- Can protect CA IDMS resources without using user exits

Terminology

- Security Domain
- Resources
- Authority
- Privileges

Security Domain

- Set of UCF and DC systems and CA IDMS local mode jobs that share a set of user definitions
- If user validation performed by external security system
  - Domain = the corporate security domain
- If user validation performed by CA IDMS Internal security
  - Domain = set of DC systems that share a user catalog (SYSUSER):

Types of resources

- Global Resources
- System Resources
- Database Resources

Global Resources

- USER
- GROUP
- USER PROFILE

Note: Definition of global resources is in user catalog (SYSUSER.DDLSEC)

System Resources

- SIGNON
- SYSTEM
- SYSTEM PROFILES
- CATEGORIZED resources
- ACTIVITY

CATEGORIZED resources:
TASK
PROGRAM
LOAD MODULE
ACCESS MODULE
RUNUNIT
QUEUE

Note: System resources are defined in the SYSTEM.DDLDML

Database Resources

- DATABASE
  - DBADMIN privilege
  - Database
  - Area
  - Rununit
  - SQL Schema
  - Non-SQL schema
  - Table
  - Access Module
- DBTABLE
- DMCL

Note:Database resources are defined in the DDLCAT/DDDML area

Resource Authorizations

Administration
- SYSADMIN
- DCADMIN
- DBADMIN

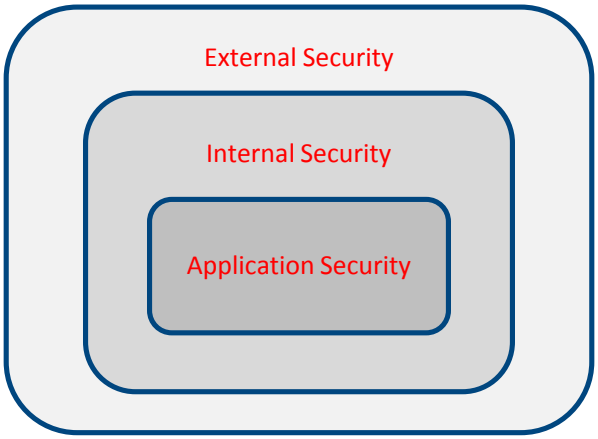
Definition
- CREATE
- ALTER
- DROP
- USE
- REFERENCE
- DISPLAY

Access
- SIGNON
- EXECUTE
- INSERT
- UPDATE
- DELETE
- DBAREAD
- DBAWRITE

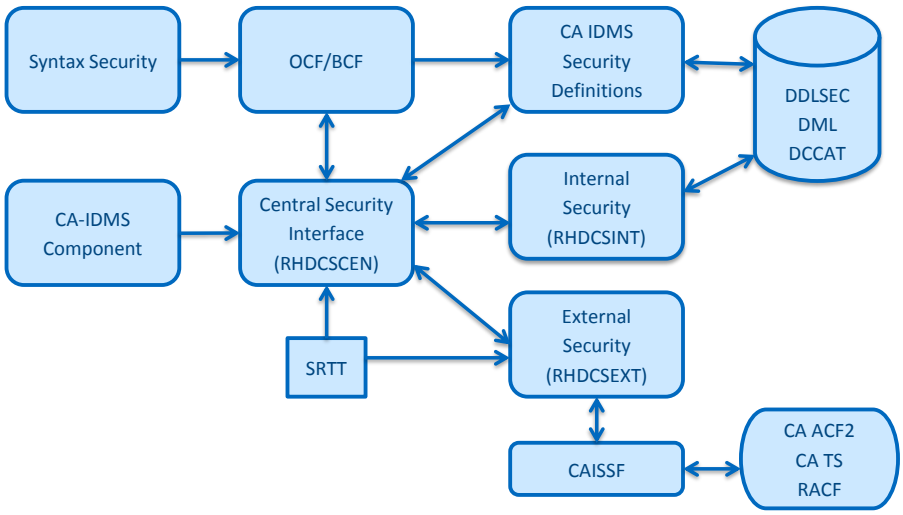
Privileges

- SIGNON
- EXECUTE
- DEFINE – CREATE, ALTER, DELETE
- REFERENCE
- USE

Architecture



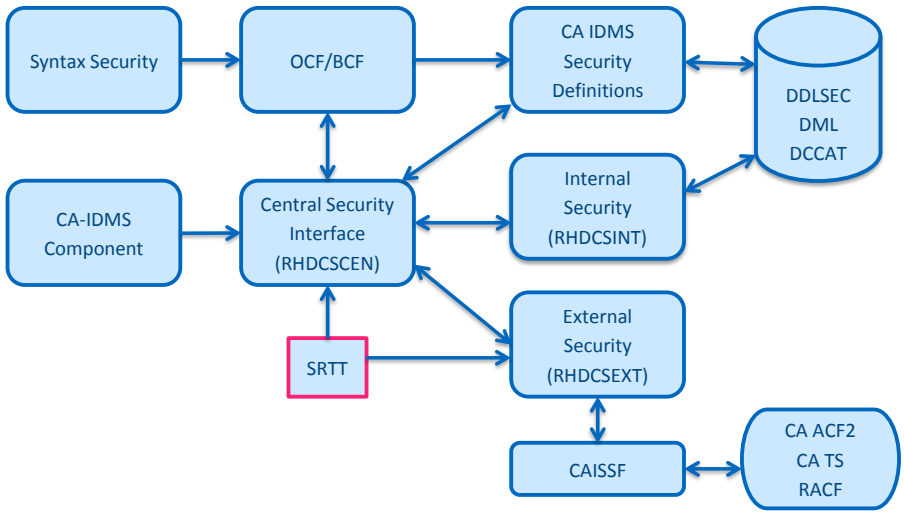
Architecture



Agenda

1. INTRODUCTION
2. SRTT TABLE
3. INTERNAL SECURITY
4. EXTERNAL SECURITY
5. SIGNON PROCESS
6. MINIMUM SECURITY RECOMMENDATIONS
7. APPLICATION SECURITY

Architecture



SRTT table

- Security Resource Type Table
- Identifies resources that are to be secured, and how they are secured (internally or externally)
- For resources secured externally, the table identifies information for the external security system
- Defined by #SRTT macros

SRTT (cont'd)

- 4 formats for #SRTT macro:
  - Initial : to denote beginning of SRTT table
  - Entry: to specify security option for all occurrences of a given resource type
  - Occurrence override (valid for type DB, SPGM, TASK)
  - Final to indicate end of SRTT table

SRTT TYPE=INITIAL

#SRTT TYPE=INITIAL
,ENVNAME=environment-name/NULL
,SGNRETN=time-interval/OFF
,SYSPROF=
,USRPROF=
,DFLTSGN=YES/NO,DFLTUID=userid/---
,EXTRUID=userid
,MAXRESN=
,SVC=svcnumber/175

#SRTT TYPE=ENTRY

RESTYPE=
,SECBY=EXTERNAL/INTernal/OFF
,EXTCLS=
,EXTNAME=(ACTIVITY/APPLname/DBName/
DDName/ENVIr/RESName/RESTYPE/
SCHEma/SSNAme/SYSTem/VERSion)

#SRTT RESTYPE=

Global Resources	TYPE= parm
SYSADMIN privilege	SYSA
User	USER
Group	GROUP
User Profile	UPRF

#SRTT RESTYPE=

System Resources	TYPE= parm
DCADMIN privilege	DCA
System	SYST
System Profile	SPRF
Signon	SGON
Activity	ACTI
Task	TASK
Load Module	SLOD
Queue	QUEU
Access Module	SACC
Program	SPGM

#SRTT RESTYPE=

Database Resources	TYPE= parm
DBADMIN privilege	DB
Database	DB
Area	DB (AREA)
Rununit	DB (NRU)
Schema (SQL)	DB (QSCH)
Non-SQL Schema	DB (NSCH)
Access Module	DB (DACC)
Table	DB (TABL)
DMCL	DMCL
Database Name Table	DBTB

#SRTT TYPE=OCCURRENCE

RESTYPE=
,RESNAME=
,SECBY=EXTERNAL/INTernal/OFF
,EXTCLS=
,EXTNAME=(ACTIVITY/APPLname/DBName/
DDName/ENVIr/RESName/RESTYPE/
SCHEma/SSNAme/SYSTem/VERSion)

#SRTT TYPE=FINAL

- That's it ! No extra parms.

SRTT examples

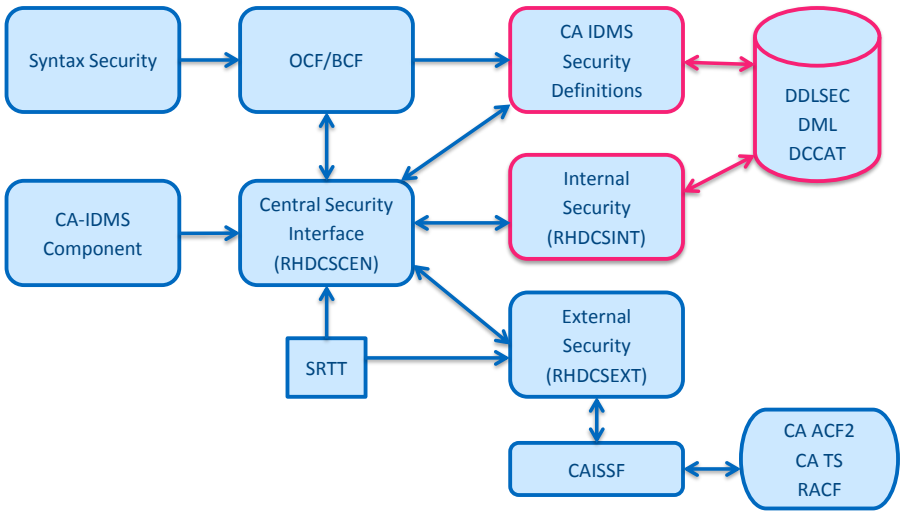
```
#SECRTT TYPE=INITIAL,SVC=210
#SECRTT TYPE=ENTRY,RESTYPE=SYSA,SECBY=INTERNAL
#SECRTT TYPE=ENTRY, RESTYPE=SGON, SECBY=EXTERNAL,        X
        EXTCLS='IDMSX', EXTNAME=RESNAME
#SECRTT TYPE=ENTRY, RESTYPE=USER, SECBY=INTERNAL
#SECRTT TYPE=ENTRY, RESTYPE=GROU, SECBY=INTERNAL
#SECRTT TYPE=ENTRY, RESTYPE=TASK, SECBY=INTERNAL
#SECRTT TYPE=OCCURRENCE, RESTYPE=TASK,                   X
        RESNAME='RHDCNP3S',SECBY=OFF
#SECRTT TYPE=ENTRY, RESTYPE=ACTI, SECBY=INTERNAL
#SECRTT TYPE=ENTRY, RESTYPE=DMCL, SECBY=INTERNAL
#SECRTT TYPE=ENTRY, RESTYPE=DBTB, SECBY=INTERNAL
```

SRTT examples (cont'd)

```
#SECRTT TYPE=ENTRY,RESTYPE=DB, SECBY=OFF
#SECRTT TYPE=OCCURRENCE, RESTYPE=DB,                    X
        RESNAME='SYSTEM',SECBY=INTERNAL
#SECRTT TYPE=OCCURRENCE, RESTYPE=DB,                    X
        RESNAME='SYSUSER',SECBY=INTERNAL
#SECRTT TYPE=OCCURRENCE, RESTYPE=DB,                    X
        RESNAME='CATSYS',SECBY=INTERNAL
#SECRTT TYPE=FINAL
```


Agenda

Architecture



Internal Security

- #SRTT RESTYPE=xxxx,SECBY=INTERNAL
- Must define the component
- Must grant privilege on component
- Report on the security

Security Global Resources - SYSADMIN

- #SRTT RESTYPE=SYSA,SECBY=INTERNAL
- No need to define SYSADMIN (it's a privilege)
- GRANT SYSADMIN TO user/group ;
- REVOKE SYSADMIN FROM user/group ;
- DISPLAY PRIVILEGE ON SYSADMIN;

Note: Do NOT grant SYSADMIN to group PUBLIC

Security Global Resources – SYSADMIN (cont'd)

```
DIS PRIVILEGES ON SYSADMIN
*+ Status = 0       SQLSTATE = 00000
*+   GRANT SYSADMIN
*+      DATE CREATED 1996-09-24-13.31.07.459530 BY ABC
*+      DATE LAST UPDATED 1996-09-24-13.31.07.459530 BY ABC
*+      TO DBA_GRP
*+      ;
*+   GRANT SYSADMIN
*+      DATE CREATED 1996-09-24-13.31.07.383594 BY ABC
*+      DATE LAST UPDATED 1996-09-24-13.31.07.383594 BY ABC
*+      TO SECURITY_GRP
*+      ;
```

Security Global Resources - USER

- #SRTT RESTYPE=USER,SECBY=INTERNAL
- CREATE/ALTER USER *userid*
  DESCRIPTION ‘user description’
  GROUP PUBLIC/*group-name*
  NAME ‘user name’
  PASSWORD “password”
  PROFILE *user-profile-name*
  ;
- DROP USER *userid-id* ;

Security Global Resources - GROUP

- #SRTT RESTYPE=GROU,SECBY=INTERNAL
- CREATE GROUP *group-name*
  DESCRIPTION ‘user description’
  ADD USER *userid1,userid2, etc*
- ALTER GROUP *group-name*
  DESCRIPTION ‘user description’
  ADD/DROP USER *userid1,userid2*
  ;
- DROP GROUP *group-name*;

Security Global Resources – USER PROFILE

- #SRTT RESTYPE=UPRF,SECBY=INTERNAL
- CREATE USER PROFILE *user-profile-name*
  ATTRIBUTES *attribute-keyword* =
  OVERRIDE=YES/NO
- ALTER USER PROFILE *user-profile-name*
  ATTRIBUTES *attribute-keyword* =
  OVERRIDE=YES/NO
  ;
- DROP USER PROFILE *user-profile-name*;

Security Global Resources – Granting access

- GRANT DEFINE/ALTER/CREATE/DISPLAY/DROP
  ON USER/GROUP/USER PROFILE */*name*
  TO *userid/group-name*
  WITH GRANT OPTION;
- REVOKE DEFINE /ALTER/CREATE/DISPLAY/DROP
  ON USER/GROUP/USER PROFILE */*name*
  FROM *userid/group-name*;

Security Global Resources – Reporting

- DISPLAY PRIVILEGES ON SYSADMIN;
- DISPLAY PRIVILEGES ON USER *userid*;
- DISPLAY PRIVILEGES ON GROUP *group-name*;
- DISPLAY PRIVILEGES ON USER PROFILE *profile-name*;

Securing Global Resources - Examples

- CREATE GROUP DBA_GROUP;
- CREATE USER VHERA00
  NAME ‘LAURA ROCHON’
  DESCRIPTION ‘HERA DBA’
  GROUP DBA_GROUP ;
- ALTER GROUP DBA_GROUP
  ADD USER VHERA01, VHERA02
  ADD USER VHERA03
  DROP USER VHERA00 ;

Securing Global Resources – Examples (cont'd)

- CREATE USER PROFILE PRODDB
  ATTRIBUTE DICTNAME=PRODDICT,
  DBNAME=PRODDB ;
- ALTER USER VHERA00
  USER PROFILE PRODDB ;
- GRANT DEFINE ON USER * TO SEC_GROUP ;
- GRANT DEFINE ON GROUP * TO SEC_GROUP ;

Security System Resources - DCADMIN

- #SRTT RESTYPE=DCA,SECBY=INTERNAL
- No need to define DCADMIN (it's a privilege)
- GRANT DCADMIN TO user/group ;
- REVOKE DCADMIN FROM user/group ;
- DISPLAY PRIVILEGE ON DCADMIN;

Note: Do NOT grant DCADMIN to group PUBLIC

Security System Resources - SYSTEM

- #SRTT RESTYPE=SYST,SECBY=INTERNAL
- CREATE RESOURCE SYSTEM *system-id ;*
- DROP RESOURCE SYSTEM *system-id*;
- DISPLAY PRIVILEGES ON SYSTEM *system-id*;

Security System Resources – SYSTEM PROFILE

- #SRTT RESTYPE=SPRF,SECBY=INTERNAL
- CREATE SYSTEM PROFILE *system-profile-name*
  ATTRIBUTES *attribute-keyword =*
  OVERRIDE=YES/NO
- ALTER SYSTEM PROFILE *system-profile-name*
  ATTRIBUTES *attribute-keyword =*
  OVERRIDE=YES/NO
  ;
- DROP SYSTEM PROFILE *system-profile-name*;

Security System Resources – SIGNON

- #SRTT RESTYPE=SGON,SECBY=INTERNAL
- Determines where user and password validation is done
- If SIGNON is NOT secured, anyone can sign on to the system using what userid they want
- SIGNON can only be granted to users, not groups

Security System Resources – SIGNON

- GRANT SIGNON ON SYSTEM *system-id*
  PROFILE *system-profile/NULL*
  TO *userid1, userid2, … ;*
- Example:
  - GRANT SIGNON ON SYSTEM SYST0020 TO VHERA00 ;
  - GRANT SIGNON ON SYSTEM SYST0010
    PROFILE SYST10-PROFILE TO VHERA00 ;
  - REVOKE SIGNON ON SYSTEM SYST0010 FROM VHERA00 ;
  - GRANT SIGNON ON SYSTEM SYST0010
    PROFILE NEW-PROFILE TO VHERA00

Security System Resources - ACTIVITY

- #SRTT RESTYPE=ACTI,SECBY=INTERNAL
- An application function becomes an activity number that is defined to the security system.
- CA ADS, DCMT, OCF/BCF, Debugger can use activity numbers to secure certain portions of their application.
- Up to 255 discrete security numbers per application
- An activity bit map is built at first security check for a function within the application (if secured internally)
- If the definition of the application does not exist, application DEFAULT will be queried.

Security System Resources – ACTIVITY (cont'd)

- DCMT activity numbers assigned thru #CTABGEN
- OCF/BCF activity numbers assigned thru #UTABGEN
- Debugger activity numbers assigned thru #GTABGEN
- CA ADS activity numbers are assigned thru ADSA

Security System Resources – ACTIVITY (cont'd)

- CREATE RESOURCE ACTIVITY *appl-name.activity-name* NUMBER *number ;*
- Statements created for application DEFAULT:
  - CREATE RESOURCE ACTIVITY DEFAULT.ACT_001 NUMBER 1;
  - CREATE RESOURCE ACTIVITY DEFAULT.ACT_002 NUMBER 2;
  - CREATE RESOURCE ACTIVITY DEFAULT.ACT_003 NUMBER 3;
    ….
  - CREATE RESOURCE ACTIVITY DEFAULT.ACT_255 NUMBER 255;

Security System Resources – ACTIVITY (cont'd)

- GRANT EXECUTE ON ACTIVITY *appl-name.activity-name* TO *userid/group-name/*PUBLIC ;
- REVOKE EXECUTE ON ACTIVITY *appl-name.activity-name* FROM *userid/group-name/*PUBLIC ;
- DISPLAY PRIVILEGES ON ACTIVITY *appl-name.activity-name;*

Security System Resources – ACTIVITY (cont'd)

```
DIS PRIVILEGES ON RESOURCE ACTIVITY DEFAULT.ACT_210 ;
*+ Status = 0       SQLSTATE = 00000
*+  GRANT EXECUTE ON ACTIVITY DEFAULT.ACT_210
*+     DATE CREATED 1996-09-24-13.31.02.781807 BY ABC
*+     DATE LAST UPDATED 1996-09-24-13.31.02.781807 BY ABC
*+     TO DBA_GRP
*+     ;
*+  GRANT EXECUTE ON ACTIVITY DEFAULT.ACT_210
*+     DATE CREATED 1998-03-31-09.25.24.163530 BY ABC
*+     DATE LAST UPDATED 2009-06-08-13.23.18.368324 BY ABC
*+     TO PGMR_GRP
*+     ;
*+  GRANT EXECUTE ON ACTIVITY DEFAULT.ACT_210
*+     DATE CREATED 2017-08-21-15.17.11.452828 BY ABC
*+     DATE LAST UPDATED 2017-08-21-15.22.20.581422 BY ABC
*+     TO SAPR2T1_GRP
*+     ;
```


Security System Resources – Categorized resources

- Categories are ONLY used in Internal Security
- Categories are used to secure tasks, programs, load modules, queues, rununits and access modules
- Maximum on 32K categories in a system
- Category bit map is built at first security check for a resource within the category
- A resource may be associated with one and only category
- Wildcarding is available in resource specification.

Security System Resources – Categorized resources (cont'd)

- Resources that can be categorized, along with SRTT RESTYPE:

Resource	#SRTT RESTYPE= Internal Security	#SRTT RESTYPE= External Security
Tasks	TASK	TASK
Programs	SPRG	SPRG
Load modules	SLOD	SLOD
Queues	QUEU	QUEU
Rununits	DB	NRU
Access Modules	DB	SACC/DACC

Security System Resources – Categorized resources (cont'd)

- Categories are ONLY used in Internal Security
- Categories are used to secure tasks, programs, load modules, queues, rununits and access modules
- Maximum on 32K categories in a system
- Category bit map is built at first security check for a resource within the category
- A resource may be associated with one and only category
- Wildcarding is available in resource specification.

Security System Resources – Categorized resources (cont'd)

- CREATE/ALTER RESOURCE CATEGORY *category-name*
  ADD/DROP
  ACCESS MODULE *dictname.schema.access-module-name*
  LOAD MODULE *dictname.Vnnnn.load-module-name*
  PROGRAM *filename.program-name*
  QUEUE *queue-name*
  RUNUNIT *dbname.subschema-name.program-name*
  TASK *task-code*
  ;

Security System Resources – Categorized resources (cont'd)

- GRANT EXECUTE ON CATEGORY *category-name*
  TO *userid/group-name*/PUBLIC ;
- REVOKE EXECUTE ON CATEGORY *category-name*
  FROM *userid/group-name*/PUBLIC ;
- DISPLAY ALL RESOURCE CATEGORY;
- DISPLAY PRIVILEGES ON RESOURCE CATEGORY
  category-name ;

Security System Resources – Categorized resources Examples

```
DISPLAY ALL RESOURCE CATEGORY AS SYN
*+ Status = 0       SQLSTATE = 00000
    DISPLAY RESOURCE CATEGORY ADS_CAT ;
    DISPLAY RESOURCE CATEGORY AGRE_CAT ;
    DISPLAY RESOURCE CATEGORY APPC_CAT ;
    DISPLAY RESOURCE CATEGORY APPROVER_CAT ;
    DISPLAY RESOURCE CATEGORY ASF_CAT ;
    DISPLAY RESOURCE CATEGORY CAT_004 ;
    DISPLAY RESOURCE CATEGORY CAT_010 ;
    DISPLAY RESOURCE CATEGORY CAT_022 ;
    DISPLAY RESOURCE CATEGORY CAT_030 ;
    DISPLAY RESOURCE CATEGORY CAT_040 ;
    DISPLAY RESOURCE CATEGORY CAT_048 ;
    DISPLAY RESOURCE CATEGORY CAT_053 ;
    DISPLAY RESOURCE CATEGORY CAT_070 ;
    DISPLAY RESOURCE CATEGORY CAT_072 ;
    DISPLAY RESOURCE CATEGORY CAT_101 ;
    DISPLAY RESOURCE CATEGORY CAT_112 ;
    DISPLAY RESOURCE CATEGORY CAT_113 ;
```


Security System Resources – Categorized resources Examples (cont'd)

```
 DISPLAY RESOURCE CATEGORY AGRE_CAT ;
*+ Status = 0       SQLSTATE = 00000
*+   CREATE RESOURCE CATEGORY AGRE_CAT
*+      DATE CREATED 1997-02-04-13.22.54.532771 BY ABC
*+      DATE LAST UPDATED 1997-02-04-13.22.54.532771 BY ABC
*+      CATEGORY NUMBER 37
*+      ADD RUNUNIT SYSTEM.QUESUB.DBP00003
*+      ADD TASK AGRE
*+      ;
    DISPLAY RESOURCE CATEGORY APPC_CAT ;
*+ Status = 0       SQLSTATE = 00000
*+   CREATE RESOURCE CATEGORY APPC_CAT
*+      DATE CREATED 1996-09-24-13.27.29.692140 BY ABC
*+      DATE LAST UPDATED 2013-02-28-14.22.41.659248 BY ABC
*+      CATEGORY NUMBER 2
*+      ADD TASK APPCDLG4
*+      ADD TASK DBDSERV1
*+      ADD TASK "06F1"
*+      ;
```

Security System Resources – Categorized resources Examples (cont'd)

```
DIS GROUP PUBLIC
*+ Status = 0       SQLSTATE = 00000
*+   GROUP "PUBLIC"
*+       GROUP IS ACTIVE
*+       DESCRIPTION 'PUBLIC Group'
*+       DATE CREATED 1996-09-24-13.27.34.390787 BY SYSTEM
*+       DATE LAST UPDATED 1996-09-24-13.27.34.390787 BY SYSTEM
*+       HOLDS EXECUTE PRIVILEGES ON CATEGORY APPC_CAT
*+       HOLDS EXECUTE PRIVILEGES ON CATEGORY UNSECURED_CAT
*+       ;
```


Security System Resources – Categorized resources Examples (cont'd)

```
DIS RESOURCE CATEGORY UNSECURED_CAT
*+ Status = 0       SQLSTATE = 00000
*+   CREATE RESOURCE CATEGORY UNSECURED_CAT
*+       DATE CREATED 1996-09-24-13.27.32.679040 BY ABC
*+       DATE LAST UPDATED 1996-09-24-13.27.32.679040 BY ABC
*+       CATEGORY NUMBER 17
*+       ADD RUNUNIT SYSTEM.IDMSSECU.DBP00001
*+       ADD TASK ADSR
*+       ADD TASK ADS2
*+       ADD TASK AGERE
*+       ADD TASK B
*+       ADD TASK BYE
*+       ADD TASK CTDI
*+       ADD TASK D
*+       ADD TASK DCMT
*+       ADD TASK DCUF
*+       ADD TASK OFF
*+       ADD TASK OPUS
*+       ADD TASK RHDCNP3S
```

Security Database Resources

- When you turn on the security option for resource type DB, several resource types are automatically secured:
  - Database
  - DBADMIN
  - Access Module
  - Area
  - Rununit
  - SQL-defined schema
  - Non-SQL-defined schema
  - Table
- Therefore, it is very important to decide whether you really need to secure a database

Security Database Resources (cont'd)

Database Resource	RESTYPE= Internal Security	RESTYPE= External Security
Database	DB	DB
DBADMIN privilege	DB	N/A
Access Module	DB	DACC
Area	DB	Area
Run unit	DB	NRU
SQL-Defined Schema	DB	QSCH
Non-SQL-Defined Schema	DB	NSCH
Table	DB	TABL

Security Database Resources (cont'd)

Database Resource	RESTYPE= Internal Security	RESTYPE= External Security
DMCL	DMCL	DMCL
DBTable	DBTB	DBTB

Security Database Resources (cont'd)

Privilege	DB	AREA	DMCL	DBTABLE
CREATE	X		X	X
ALTER	X		X	X
DROP	X		X	X
DISPLAY	X		X	X
USE	(1)	(1)	X	X
DBAREAD		X		
DBAWRITE		X		
DBADMIN	X			

(1) Privilege application only to non-SQL-defined databases

Security Database Resources (cont'd)

- When a program issues a BIND RUN-UNIT or a CONNECT statement, the specified dbname can be either an actual dbname (in the DBTable) or a segment name.
- DBMS will search DBTable for DBNAME specified for BIND RUN-UNIT. If found, will look for areas in segments tied to that dbname in DBTABLE. If not found, the DBMS will look for segment name in DMCL matching dbname. If found, all areas must be in that segment. If not found, error.

Security Database Resources - DMCL

- #SRTT RESTYPE=DMCL,SECBY=INTERNAL
- Securing DMCL determines who can create/modify it & execute utilities against DMCL journal files
- GRANT DEFINE/ALTER/CREATE/DISPLAY/DROP/USE
  ON DMCL *dmcl-name*
  TO *userid/group-name*/PUBLIC
  WITH GRANT OPTION
  ;
- REVOKE DEFINE/ALTER/CREATE/DISPLAY/DROP/USE
  ON DMCL *dmcl-name*
  FROM *userid/group-name*/PUBLIC ;

Security Database Resources - DBTABLE

- #SRTT RESTYPE=DBTB,SECBY=INTERNAL
- Securing the DBTABLE in order to maintain database security that is based on occurrence overrides
- GRANT DEFINE/ALTER/CREATE/DISPLAY/DROP/USE
  ON DBTABLE *database-table-name*
  TO *userid/group-name*/PUBLIC
  WITH GRANT OPTION
  ;
- REVOKE DEFINE/ALTER/CREATE/DISPLAY/DROP/USE
  ON DBTABLE *database-table-name*
  FROM *userid/group-name*/PUBLIC ;

Security Database Resources - DB

- #SRTT RESTYPE=DB,SECBY=INTERNAL
- When specifying DB, securing access to specific database
- GRANT DEFINE/ALTER/CREATE/DISPLAY/DROP/USE
  ON DB *database-name*
  TO *userid/group-name*/PUBLIC
  WITH GRANT OPTION
  ;
- REVOKE DEFINE/ALTER/CREATE/DISPLAY/DROP/USE
  ON DB *database-name*
  FROM *userid/group-name*/PUBLIC ;

Security Database Resources - DBADMIN

- #SRTT RESTYPE=DB,SECBY=INTERNAL
- When specifying DB, securing access to specific database
- GRANT DBADMIN ON DB *database-name*
  TO *userid/group-name*/PUBLIC ;
- REVOKE DBADMIN ON DB *database-name*
  FROM *userid/group-name*/PUBLIC ;

Security Database Resources - AREA

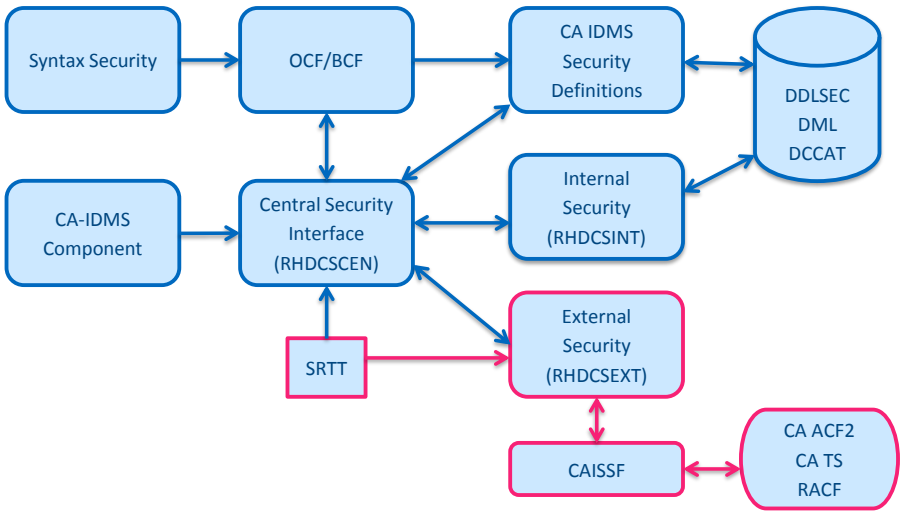
- #SRTT RESTYPE=DB,SECBY=INTERNAL
- When specifying DB, securing access to areas for that DB
- GRANT DBAREAD/DBAWRITE/USE
  ON AREA *area-name*
  TO *userid/group-name*/PUBLIC
  WITH GRANT OPTION
  ;
- REVOKE DBAREAD/DBAWRITE/USE
  ON AREA *area-name*
  FROM *userid/group-name*/PUBLIC ;

Security Database Resources - Examples

```
DIS GROUP DBA_GRP ;
*+ Status = 0       SQLSTATE = 00000
*+  CREATE GROUP "DBA_GRP"
*+      GROUP IS ACTIVE
*+      DATE CREATED 1996-09-24-13.27.33.313375 BY ABC
*+      DATE LAST UPDATED 2017-03-15-10.24.31.233913 BY ABC
…
*+      HOLDS SYSADMIN PRIVILEGES
*+      HOLDS DEFINE PRIVILEGES ON DMCL *
*+      HOLDS DEFINE PRIVILEGES ON DBTABLE *
*+      HOLDS DEFINE PRIVILEGES ON DB SYSTEM
*+      HOLDS DBADMIN PRIVILEGES ON DB SYSTEM
*+      HOLDS USE PRIVILEGES ON AREA SYSTEM.*
*+      HOLDS DBAREAD, DBAWRITE PRIVILEGES ON AREA SYSTEM.*
*+      HOLDS USE PRIVILEGES ON AREA SYSUSER.*
*+      HOLDS DBAREAD, DBAWRITE PRIVILEGES ON AREA SYSUSER.*
*+      HOLDS USE PRIVILEGES ON AREA CATSYS.*
*+      HOLDS DBAREAD, DBAWRITE PRIVILEGES ON AREA CATSYS.*
```


Agenda

Architecture



Supported External Security Systems

- CA ACF2
- CA TopSecret
- RACF (IBM)

External Security Implementation

- #SRTT RESTYPE=xxxx,SECBY=EXTERNAL
- Must specify EXTCLAS and EXTNAME on #SRTT TYPE=ENTRY
- EXTNAME can have multiple keywords to create the name passed to the External Security Package
- Order of keywords specified in EXTNAME must match in same order within the External Security Package

External Resource Name – Global Resources

Resource	RESNAME	RESTYPE	Other available keywords
SYSADMIN	@RESERVED@	SYSA	
User	*Userid*	USER	
Group	*Group-id*	GROU	
User Profile	*Profile-name*	UPRF	

External Resource Name – System Resources

Resource	RESNAME	RESTYPE	Other available keywords
DCADMIN	@RESERVED@	DCA	
System	*System-id*	SYST	
Signon	*System-id*	SGON	
System Profile	*Profile-name*	SPRF	
Activity	*Application-name*	ACTI	APPLname, ACTIvity
Task	*Task-code*	TASK	SYSTem

External Resource Name – System Resources

Resource	RESNAME	RESTYPE	Other available keywords
Load Module	*Load-module-name*	SLOD	DBName, VERSion
Queue	*Queue-name*	QUEU	SYSTem
Access Module	*Access-module-name*	SACC	DBNAme, SCHEma
Program	*Program-name*	SPGM	SYSTem, DDName

External Resource Name – Database Resources

Resource	RESNAME	RESTYPE	Other available keywords
Database	*Database-name*	DB	
Area	*Area-name*	AREA	DBNAme
Rununit	*Program-name*	NRU	DBNAme, SSNAme
SQL Schema	*Schema-name*	QSCH	DBNAme
Non-SQL Schema	*Non-SQL-schema-name*	NSCH	DBNAme, VERSion

External Resource Name – Database Resources

Resource	RESNAME	RESTYPE	Other available keywords
Access Module	*Access-module-name*	DACC	DBNAme, SCHEma
Table	*Table-name*	TABL	DBNAme, SCHEma
DMCL	*DMCL-name*	DMCL	
Database Name Table	*Database-table-name*	DBTB	

External Security Examples

```
#SECRTT TYPE=INITIAL, ENVNAME=IDMS, SGNRETN=60,          X
              SVCNUM=225
#SECRTT TYPE=ENTRY,RESTYPE=TASK,SECBY=OFF,               X
              EXTCLS='TSK',EXTNAME=(ENVIR,RESNAME)
#SECRTT TYPE=OCCUR,RESTYPE=TASK,SECBY=EXT,               X
              RESNAME='ADS'
#SECRTT TYPE=OCCUR,RESTYPE=TASK,SECBY=EXT,               X
              RESNAME='ADSA'
#SECRTT TYPE=OCCUR,RESTYPE=TASK,SECBY=EXT,               X
              RESNAME='ADSAT'
#SECRTT TYPE=OCCUR,RESTYPE=TASK,SECBY=EXT,               X
              RESNAME='ADSC'
#SECRTT TYPE=OCCUR,RESTYPE=TASK,SECBY=EXT,               X
              RESNAME='ADSCT'
'
```


External Security Examples

```
ACF75052 RESOURCE RULE IDMS STORED BY ABC
ON 12/29/17-02:36
 $KEY(IDMS) TYPE(TSK)
  ADSR UID(*) ALLOW
  ADSRT UID(*) ALLOW
  ADS2 UID(*) ALLOW
  ADS2T UID(*) ALLOW
  DMLO UID(ISD*********TS) ALLOW
  IDD UID(************TSDBCB) ALLOW
```

Agenda

1. INTRODUCTION
2. SRTT TABLE
3. INTERNAL SECURITY
4. EXTERNAL SECURITY
5. SIGNON PROCESS
6. MINIMUM SECURITY RECOMMENDATIONS
7. APPLICATION SECURITY

SIGNON Process

- If SRTT has RESTYPE=SGON, SECBY=OFF
  - Signon is unsecured, NO password validation
- If SRTT has RESTYPE=SGON, SECBY=INTERNAL
  - Signon is secured, password validation is performed by Internal Security System
- If SRTT has RESTYPE=SGON, SECBY=EXTERNAL
  - Signon is secured, password validation is performed by Extenal Security System

SIGNON Options Mixed

- If SRTT has some components secured internally, and some other components secured externally, then a signon is performed by both Security Systems, but the password validation is performed by the System identified by the RESTYPE=SGON,SECBY entry.

Types of SIGNON

- An Explicit SIGNON is when
  - SIGNON or S tasks
  - Linking to RHDCSNON
- An Automatic SIGNON is when
  - User has already signed on to another system (like TSO, CICS)
  - No password validation
- A Default SIGNON is when
  - DFLTSGN=YES and DFLTUID specified on SRTT
  - User is not signed on, and a security check is performed

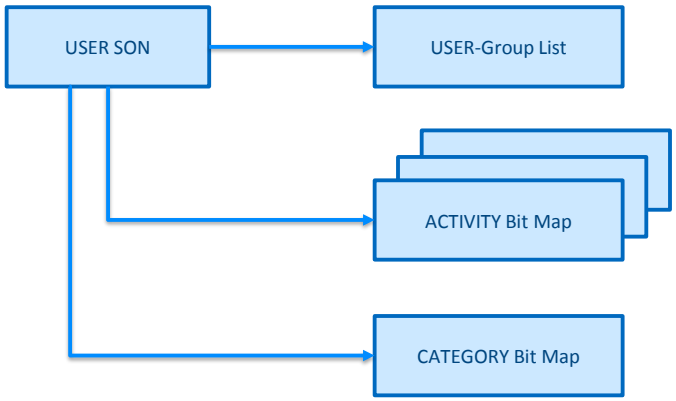
SIGNON Processing Flow

- Identify userid
- In DC/UCF:
  - If a user is already signed on to terminal, sign the user off
  - If the user signing on to an interactive terminal is already signed on to another interactive terminal, deny the signon unless MULTIPLE SIGNON IS ALLOWED
- Validate the user & password
- Validate user's access to system
- In DC/UCF, update user's password if requested (explicit signon only)
- Build the group list for user

SIGNON Processing Flow (cont'd)

- Build the session profile from user profile information and possibly system profile information
- Invoke the CLIST identified by the CLIST attribute if one exists in session profile

SIGNON Control Block



Agenda

1. INTRODUCTION
2. SRTT TABLE
3. INTERNAL SECURITY
4. EXTERNAL SECURITY
5. SIGNON PROCESS
6. MINIMUM SECURITY RECOMMENDATIONS
7. APPLICATION SECURITY

Minimum Security Recommendations

- SGON should be secured
- SYSTEM, CATSYS, SYSUSER should be secured
- Any TASK code that can affect the system

Securing Dictionary

```
#SECRTT TYPE=ENTRY,RESTYPE=DB,SECBY=OFF
#SECRTT TYPE=OCCURRENCE,RESTYPE=DB,
        RESNAME='SYSTEM',SECBY=INTERNAL
#SECRTT TYPE=OCCURRENCE,RESTYPE=DB,
        RESNAME='CATSYS',SECBY=INTERNAL
#SECRTT TYPE=OCCURRENCE,RESTYPE=DB,
        RESNAME='SYSUSER',SECBY=INTERNAL
```

Securing Dictionary (cont'd)

- Granting Access to Security Catalog (SYSUSER):
  - GRANT DBAREAD/DBAWRITE ON AREA SYSUSER.DDLSEC TO GROUP_SEC ;
  - GRANT DBADMIN ON DB SYSUSER TO GROUP_SEC;
  - GRANT USE ON NONSQL SCHEMA IDMSSECU TO GROUP_SEC;
  - GRANT DISPLAY ON NONSQL SCHEMA IDMSSECU TO GROUP_SEC;

Agenda

Application Security

- CA IDMS DML Online (DMLO)
- CA OLQ
- CA Culprit
- IDD
- CA ADS applications

CA IDMS DML Online (DMLO)

- Multiple levels of security for DMLO:
  - 1 : No security check is performed.
  - 2 : DMLO verifies user/password are valid in requested dictionary
  - 3 : DMLO verifies user/password are valid in requested dictionary and that user has access to requested subschema
- Access restrictions
  - Restricting Usage Mode Access Globally
  - Restricting Usage Mode Access by User

DMLO (cont'd)

- DMLO can be secured within a dictionary by using following IDD syntax:

  ADD PROGRAM DBMSDMLO VERSION IS n

  Where n is the security level 1, 2 or 3

DMLO (cont'd)

- User has access to a given subschema with the following IDD syntax:

  MOD USER userid

  PASSWORD password

  INCLUDE ACCESS TO SUBSCHEMA ssc-name OF

  SCHEMA schema-name V schema-version

  .

DMLO (cont'd)

- Area usage mode can be secure globally for DMLO by using following IDD syntax

MOD PROGRAM DBMSDMLO V n
PROGRAM DESCRIPTION IS 'valid usage modes'.

Valid usage modes : SR, SU, PR, PU, ER , EU

DMLO (cont'd)

- Area usage mode can be secured for a user by using following IDD syntax :

MOD USER userid
PASSWORD password
USER DESCRIPTION IS 'valid usage modes'
.

CA OLQ

- Security for CA OLQ is turned ON or OFF within the dictionary with the following command:

```
SET OPTIONS FOR DICTIONARY
    SECURITY FOR OLQ IS ON.
```

- Only user who are defined in the dictionary will have access to OLQ

```
MOD USER userid AUTHORITY FOR UPDATE IS OLQ .
```


CA OLQ (cont'd)

- Following parameters on USER statement in IDD further qualify what USER can do with OLQ:
  - ACCESS TO QFILE
  - ACCESS TO QFILE FIRSTEN
  - OLQ QFILE IS (NOT) ALLOWED/ONLY
  - OLQ MRR IS (NOT) ALLOWED
  - OLQ MANDATORY/OPTIONAL INTERRUPT
  - OLQ SORT IS (NOT) ALLOWED
  - OLQ QFILE SAVE IS (NOT) ALLOWED
  - OLQ MENU-MODE IS (NOT) ALLOWED/ONLY
  - OLQ ACCESS IS OLQ/ IDMS SQL

CA OLQ (cont'd)

- Following parameters on USER statement in IDD set up defaults for user in OLQ:
  - OLQ DEFAULT OPTION IS HEADER
  - OLQ DEFAULT OPTION IS (NO) ECHO
  - OLQ DEFAULT OPTION IS ALL
  - OLQ DEFAULT OPTION IS (NO) FILLER
  - OLQ DEFAULT OPTION IS (NO) INTERRUPT
  - OLQ DEFAULT OPTION IS WHOLE/PARTIAL
  - OLQ DEFAULT OPTION IS FULL/SPARSE
  - OLQ DEFAULT OPTION IS (NO) OLQ HEADER

CA OLQ (cont'd)

- Following parameters on USER statement in IDD set up defaults for user in OLQ (cont'd) :
  - OLQ DEFAULT OPTION IS (NO) COMMENTS
  - OLQ DEFAULT OPTION IS (NO) PATH STATUS
  - OLQ DEFAULT OPTION IS (NO) CODE TABLE
  - OLQ DEFAULT OPTION IS (NO) EXTERNAL PICTURE
  - OLQ DEFAULT OPTION IS VERBOSE/TERSE

CA Culprit

- CA Culprit security is established at multiple levels:
  - Installation security
  - Product security
  - User security
  - Auto attribute security

CA Culprit – Installation security

- CA Culprit is installed with security either ON or OFF (the default) during the CA IDMS install
- The CA Culprit report listing contains a message indicating

  INSTALLATION SECURITY OPTION IS NO

- If security is enabled, Culprit will automatically check the data dictionary to determine the security level in effect

CA Culprit – Product security

- Security for Culprit is turned ON or OFF within the dictionary with the following command:

```
SET OPTIONS FOR DICTIONARY
    SECURITY FOR CULPRIT IS ON.
```

- Only authorized users will be able to run Culprit jobs

CA Culprit – User security

- The following IDD clause specifies that only users with CULPRIT AUTHORITY can authorize other users to access files and subschemas to run Culprit reports:

```
ADD USER userid
    INCLUDE AUTHORITY FOR UPDATE IS CULPRIT.
```

CA Culprit – User security (cont'd)

- User has access to a given subschema or file with the following IDD syntax:

```
MOD USER userid
    INCLUDE ACCESS TO SUBSCHEMA ssc-name OF
        SCHEMA schema-name V schema-version .

MOD USER userid
    INCLUDE ACCESS TO FILE file-name .
```


CA Culprit – User security (cont'd)

- Users can make changes to record layouts and file definitions if assigned the OVERRIDES clause:

```
ADD USER userid
    CULPRIT OVERRIDES ARE (NOT) ALLOWED.
```

CA Culprit – Auto Attribute Security

- If the following SET OPTIONS FOR DICTIONARY clause is set, Culprit will automatically generate REC parameters & retrieves characteristics of a file defined to IDD:

  SET OPTIONS FOR DICTIONARY
  AUTO ATTRIBUTES ARE ON.

- If set to OFF, Culprit will automatically generate REC parameters, and field definition

IDD

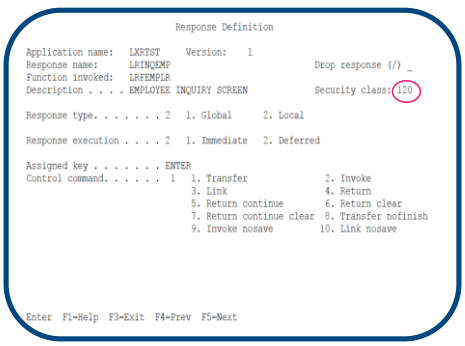
- SET OPTIONS FOR DICTIONARY
  - AUTHORIZATION IS ON/OFF
  - SECURITY FOR IDD SIGNON IS ON/OFF
  - USER SIGNON OVERRIDE IS ON/OFF
  - INDIVIDUAL PASSWORD SECURITY OVERRIDE IS ON/OFF
- Entity Security
  - RESPONSIBLE FOR ALL

CA ADS Application Security

- The CA ADS application compiler provides 2 security features to define security within the application:
  - Security for responses
  - Signon Security

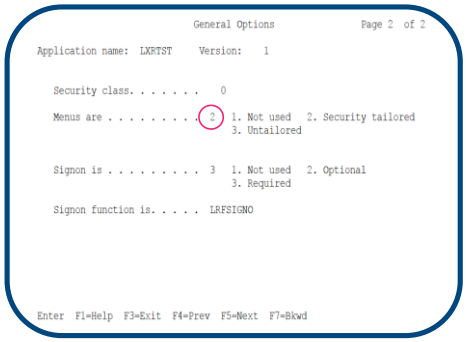
CA ADS Response Security

- To implement Response Security, you enter a number in the **Security Class** field in the ADSA Response Definition screen



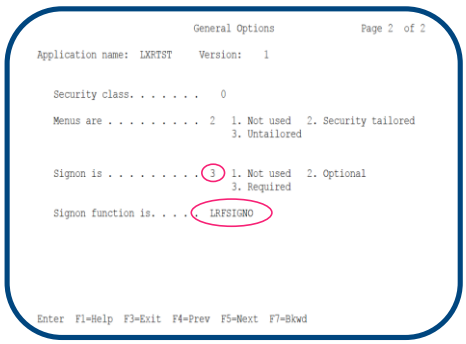
CA ADS Response Security (cont'd)

- If menus are Security tailored, page 2 of General Options screen within ADSA, menu will only contain list of functions user has access to.



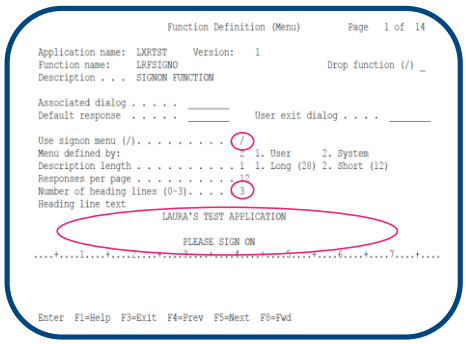
CA ADS Signon Security

- To implement Signon Security, specify Signon is either Optional or Required on page 2 of the General Option Screen, along with a signon Function name:



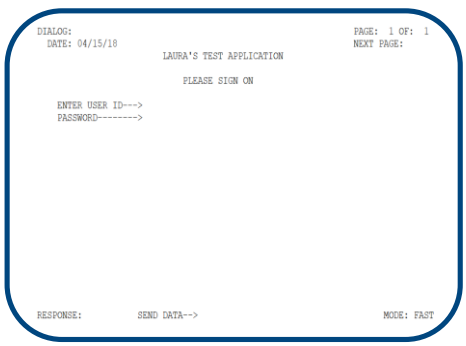
CA ADS Signon Security (cont'd)

- Define the Signon Function



CA ADS Signon Security (cont'd)

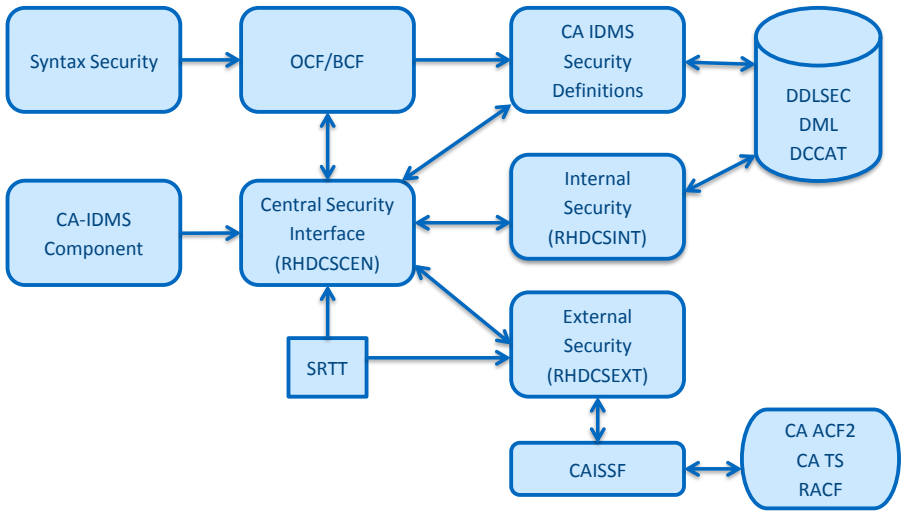
- At runtime, CA ADS will generate the signon screen:



Agenda

1. INTRODUCTION
2. SRTT TABLE
3. INTERNAL SECURITY
4. EXTERNAL SECURITY
5. SIGNON PROCESS
6. MINIMUM SECURITY RECOMMENDATIONS
7. APPLICATION SECURITY

Architecture



Summary

- You can secure basically anything you want in CA IDMS
- There is a cost to security, therefore must decide what you want to secure
- Securing entity type DB means that the database is secured at many levels. If access to DB is high, high cost.
- SIGNON should be secured
- System dictionaries should be secured

Additional information

KB000025506
https://comm.support.ca.com/kb/security-definitions-for-task-codes-in-idms-central-version/kb000025506

KB000026074
https://comm.support.ca.com/kb/implementing-signon-security-in-an-idms -central-version/kb000026074

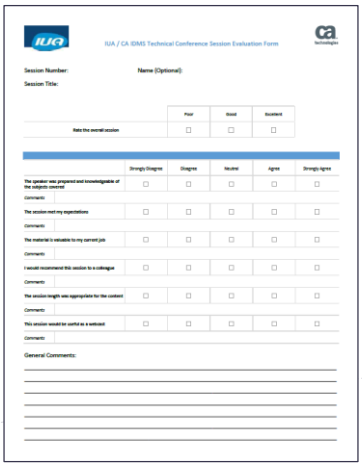
KB000025870
https://comm.support.ca.com/kb/how-to-secure-application-responses-in-idms/kb000025870

KB000077716
https://comm.support.ca.com/kb/is-there-a-need-for-user-defs-in-idms-security-dictionary-when-all-resources-are-secured-externally/kb000077716

Questions & Answers

Please Complete a Session Evaluation Form

- The number for this session is **P04**
- After completing your session evaluation form, place it in the envelope at the front of the room


IUA / CA IDMS Technical Conference Session Evaluation Form

Session Number: Name (Optional):

Session Title:

	Poor	Good	Excellent
Rate the overall session	☐	☐	☐

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The speaker was prepared and knowledgeable of the subjects covered	☐	☐	☐	☐	☐
Comments:					
The session met my expectations	☐	☐	☐	☐	☐
Comments:					
The material is valuable to my current job	☐	☐	☐	☐	☐
Comments:					
I would recommend this session to a colleague	☐	☐	☐	☐	☐
Comments:					
The session length was appropriate for the content	☐	☐	☐	☐	☐
Comments:					
This session would be useful as a webcast	☐	☐	☐	☐	☐
Comments:					

General Comments: