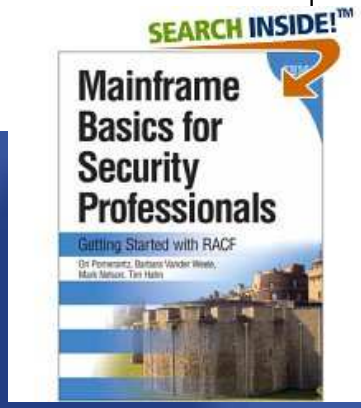
RACF® z/OS® V1.12 Update

Tivoli® zSecure™ Users Group
August 2011



Mark Nelson, CISSP®, CSSLP®
z/OS® Security Server (RACF) Design and Development
IBM Poughkeepsie
markan@us.ibm.com

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Agenda

A Quick Review of what was new with V1.11 RACF

What's new with z/OS V1.12 RACF?

- Generic Profile Loading
- SAFTRACE filtering by user ID or general resource class
- "Ghost" Generics
- Caller's Address in RACXTRT work area
- Support for ICSF

What's new with z/OS V1.12 Digital Certificate Support?

- Support for elliptic curve cryptography (ECC)
- Longer RSA keys
- DSA key types
- Extended certificate validity
- Certificate Management protocol support
- Configurable maintenance window

DB2 for z/OS Version 10

Statements of Direction for z/OS V1.13 RACF

A Quick Review of what was New in V1.11 RACF

z/OS V1.11 RACF

New with z/OS V1.11 RACF:

- Program Object Signature Verification
- Logon Statistics Suppression
- Identity Propagation
- R_admin extract for General Resource
- LDAP Change logging for General Resources
- IRRXUTIL: REXX Interface to Extract Information from the RACF DB
- Automatic assignment of UID and GID to users of Unix System Services
- Profile name in Authorization Exits
- IRRADU00 support for WAS and TKLM
- Unique indicator in ACEE control block (ACEE' vs 'acee') once the ACEE has been deleted
- RACDCERT multi-byte Character Improvements
- PKI Key Pair Generation and Certificate Recovery
- PKI Web Pages through Java Server Page (JSP)
- PKI Support for SHA256 with RSA Signature Algorithm

z/OS V1.12

z/OS V1.12: Generic Profile Load Performance

- **RACF caches up to 4 sets of generic profile names per address space to speed up authorization checks for resources which are covered by generic profiles.**
  - Known as **GATE**s (**G**eneric **A**nchor **T**able **E**ntries).
  - One per data set HLQ or general resource class that is neither SETROPTS RACLISTed, RACLISTed using RACROUTE REQUEST=LIST,GLOBAL=YES, or SETROPTS GENLISTed
- **If an address space uses more than 4 sets of profiles RACF discards the least recently used list of generic profiles**
- **If a deleted HLQ or class is referenced, the list is built again, which can result in thrashing**
- **Prior to V1.12 what could you do?**
  - Split the RACF database
  - Physically rename data sets to reduce the number of generic profiles under a single HLQ
  - Doing an analysis of the existing generic profiles to try to reduce their numbers
  - Implementing a RACF Naming Convention Table

z/OS V1.12: Generic Profile Load Performance…

- **With V1.12, you can configure the number of sets of profiles!**
  - Specified using the RACF SET command
  - Can be set system wide or by job name
  - Minimum: 4; Maximum: 99
  - A new TRACE operand has been added to the SET command to capture data about the caching of generic profiles to assist IBM support in diagnosing problems.
- **RACF has reorganized the way that GATEs are processed:**
  - Now in 64-bit storage (instead of ELSQA)
  - No longer searched linearly

z/OS V1.12: Generic Profile Load Performance…

- **The SET Command:**

```
SET …
        [ GENERICANCHOR(
        {SYSTEM | JOBNAME(jobname ...) }
        {COUNT( number ) | RESET })
        ]
```

- **SYSTEM** increases the number of generic profile caches system wide, for all jobs which do not have an overriding value
- **JOBNAME i**ncreases the number of generic profile caches for all jobs which match the value specified. “*” may be used as a “don’t care” character at the end
- Additional tracing can be activated using this SET command:
  - **SET … [TRACE (…{ GENERICANCHOR...})**

z/OS V1.12: Generic Profile Load Performance…

- **The SET LIST Command shows the status:**

```
RACFR12 IRRH005I (@) RACF SUBSYSTEM INFORMATION:
TRACE OPTIONS - IMAGE
- NOAPPC
- SYSTEMSSL
- RACROUTE
  2 5 9
- NOCALLABLE
- NOPDCALLABLE
- NODATABASE
- GENERICANCHOR (or NOGENERICANCHOR)
...
PASSWORD SYNCHRONIZATION IS *NOT* ALLOWED
AUTOMATIC DIRECTION OF APPLICATION UPDATES IS *NOT*
ALLOWED
GENERICANCHOR:
 SYSTEM: COUNT(nn)
 JOBNAME: job1 COUNT(nn)
 job2* COUNT(nn)
```

z/OS V1.12: Generic Profile Load Performance…

- **ICHEINTY macro now supports returning multiple generic profile names:**
  - New keyword: INDEX=MULTIPLE
  - Must come from the same L1 index block
  - Must have the same HLQ or class name
  - Returned in a workarea that is provided by the caller which must be at least 4K in size
    - Each profile that is returned is prefixed with a one-byte length

z/OS V1.12: SAFTRACE Filtering

- **The SAFTRACE facility, allows an in-depth analysis of the calls made from resource managers to RACF**
  - Can trace at the RACROUTE, callable service, or ICHEINTY level
  - Cannot instruct SAFTRACE to only trace for a specific class or specific user ID
  - Trace records are written to GTF and formatted with IPCS
  - Intended for use under the direction of RACF's service team
  - SET Syntax:

```
SET TRACE( …
           ASID(asid ... |*) | NOASID | ALLASIDS
           JOBNAME(jobname ... |*) | NOJOBNAME | ALLJOBNAMES
           RACROUTE(ALL | NONE | TYPE(type ...)) | NORACROUTE
           …
          )
```

z/OS V1.12: SAFTRACE Filtering by Class

- **With V1.12, you can control SAFTRACE records for RACROUTE and database (ICHEINTY) access by class:**

```
SET TRACE(CLASS(class … |*)      |
          IFCLASS(class …|*)     |
          NEVERCLASS(class …|*) |
          NOCLASS)
```

- **CLASS is an inclusive setting**
  - Trace records which match CLASS are recorded.
  - Trace records which do not match CLASS() *may* be recorded if they match another setting, like ASID or JOBNAME.
- **IFCLASS is an exclusive setting**
  - Trace records which do not match IFCLASS are always discarded, even if they match other trace setting, like ASID or JOBNAME.
- **NEVERCLASS** discards all trace records whose class names match, regardless of other settings

z/OS V1.12: SAFTRACE Filtering by User ID

- **With V1.12, you can also control SAFTRACE records created for RACROUTE traces by user ID:**

```
SET TRACE(USERID(userid ... |*) |
          IFUSERID(userid …|*) |
          NEVERUSERID(userid …|*) |
          NOUSERID)
```

- **USERID is an inclusive setting:**
  - Trace records which match the user id are recorded
  - Trace records which do not match USERID() *may* be recorded if they match another setting, like CLASS or JOBNAME
- **IFUSERID is an exclusive setting:**
  - Trace records which do not match IFUSERID are always discarded.even if they match other trace setting, like CLASS or JOBNAME
- **NEVERUSERID** discards all trace records whose user id names match, regardless of other settings

z/OS V1.12: “Ghost” Generics

- **RACF requires that SETROPTS GENERIC is in effect for a class before generic profiles are defined in the class**
  - If not, the profile is created as a discrete profile which contains generic characters, such as “*”, “&”, or “%”
  - Profiles such as these are:
    - Not involved in access control decisions
    - Not what you intended
    - Displayed by SEARCH, RLIST, and LISTDSD without the “(G)” after the name
    - Require that you turn generics and GENCMD off for the class, delete the profile, SETROPTS GENERIC the calss (which also turns GENCMD on), and redfine the profile
    - Annoyances to security administrators, systems programmers, and auditors

z/OS V1.12: “Ghost” Generics…

- **With V1.12, RACF now issues a warning message when creating a profile which contains generic characters (*,% or &) in a non-generic class**

```
ICH10321I The profile name profile_name contains
genericcharacters, but generics are not enabled for class
class_name.  A discrete profile has been created.
```

- **The message is suppressed for profiles in the RACFVARS class, in which discrete profiles with generic characters are intentionally created**

z/OS V1.12: “Ghost” Generics…

- **The RLIST command does not show existing ghost generic profiles, unless '*' is specified for the profile name**
- **The SEARCH command does display ghost generic profiles**
- **Both commands will now label ghost generic profiles as '(UNUSABLE)' in their output**

```
RLIST FACILITY T*

  CLASS      NAME
  -----      ----
  FACILITY   T* (UNUSABLE)   <--- ghost generic indicator
   ...

   CLASS NAME
   ----- ----
   FACILITY T* (G)  <-- Standard generic indicator
   ...
```

z/OS V1.12: “Ghost” Generics…

- **NOGENERIC keyword added to the RDELETE command to facilitate the deletion of ghost generic**

```
RDELETE FACILITY T* NOGENERIC
```

- **Specifies that you want RACF to delete the discrete profile**
  - If a generic profile with the same name exists, it will be unaffected.
- **SAF callable service R_admin also updated such that the Delete function supports a GENERIC=N flag**
- **RACF panels also support NOGENERIC processing**

z/OS V1.12: Caller's Address in EXTRACT Area

- **Applications for which RACF gets storage on a RACROUTE REQUEST=EXTRACT are required to free this storage when the application is finished with the data**
- **Ill-behaved applications which don't free this area can cause an out-of-storage condition**
  - It's difficult to identify the offending application/request as there is no information which ties the application to the storage
- **With V1.12, the callers ASID and return address are placed in the returned work area to assist in identifying the application which create the REQUEST=EXTRACT work area**
- **Mapped in EXTWKEA in IRRPRXTW**

z/OS V1.12: RACF Enhancement for ICSF

- **New keyword on ICSF segment on CSFKEYS, GCSFKEYS, XCSFKEY, and GXCSFKEY profiles allows the specification of controls on high performance secure keys**
  - **SYMCPACFWRAP([YES|NO])** Can this High Performance Secure key be exported?

z/OS V1.12: Digital Certificate Enhancements

- **Support for elliptic curve cryptography (ECC) when creating certificates and when processing certificates created using ECC**
  - Complete SHA2 support (SHA224, SHA256, SHA384, SHA512)
  - Support for RACDCERT BIND and IMPORT on ECC and DSA keys
  - Support for ECC certificates and ECC keys from RACF key rings and PKCS#11 tokens using the R_datalib callable service
- **Support for creating RSA keys up to 4096 bits**

z/OS V1.12: Digital Certificate Enhancements…

- **Support for long issuer distinguished names**
  - Current limitation is 246 characters for the issuer's distinguished name
  - Supported by RACDCERT ADD and GENCERT, R_datalib, InitACEE, and PKI Services
  - Rolled back to z/OS V1.10 and V1.11
    - RACF APAR: OA30560
    - PKI APAR: OA30952
- **Extend certificate validity date beyond its current limit** (PKI Services:2038, RACF:2041)
  - RACF: Until the year 9999, PKI Services: For 9999 days)
  - Supported by RACDCERT ADD, IMPORT, GENCERT, REKEY, LIST, and CHECKCERT and PKI Services
  - Rolled back to V1.10 add V1.11
    - RACF APAR: OA30560 (except RACDCERT GENCERT and REKEY)
    - PKI APAR: OA30952 (requires LE PTF UK47654 (V1.10), UK47655 (V1.11)

z/OS V1.12: PKI Services Enhancements

- **Support for certificate management protocol (CMP)**
  - CMP is the protocol that is used to manage X.509 certificates within a PKI-infrastructure. The support of these CMP in accordance with RFC 4210/4211 allows greater interoperability of z/OS PKI Services:
    - Certificate Request Message, type 2 (cr)
    - Certificate Response Message, type 3 (cp)
    - PKCS10 Certificate Request Message, type 4 (p10cr)
    - Revocation Request Message, type 11 (rr)
    - Revocation Response Message, type 12 (rp)
    - Error Message, type 23 (error)
- **Support for custom X.509 certificate extensions**
- **Support for the posting of certificates and certificate revocation lists (CRLs) to LDAP at any time**
- **Configurable maintenance task execution time**
- **For more detail, see session AST07, "PKI Services, the Best Kept Secret in z/OS", Wai Choi, Wednesday, 10:45 AM.**

RACF Access Control Module for DB2 for z/OS Version 10

DB2 V10: New DB2 System Authorities

- **DB2 for z/OS Version 10 introduces new system authorities that allow for a finer granularity of control:**
  - **SECADM:** Manage all of the security-related objects in DB2 and control access to all DB2 resources in native DB2 security
  - **System DBADM:** Manage most objects in a DB2 subsystem, without having the ability to access data or control access to data
  - **DATAACCESS:** Access data in all user tables, materialized query tables, and views and execute plans, packages functions and procedures in a DB2 subsystem.
  - **ACCESSCTRL:** Grant all authorities and privileges except, DBADM, DATAACCESS, ACCESSCTRL and privileges on security-related objects.
  - **SQLADM:** Monitor and tune DB2 without have any other privilege

DB2 V10: New DB2 System Authorities…

- **If you are using the RACF Access Control Module for DB2 (DSNXRXAC) you can grant these authorities by giving a user READ authority to these resource names in the indicated class:**

DB2 Authority	RACF General Resource Class	Resource Name
ACCESSCTRL	DSNADM	*db2-subsystem*.ACCESSCTRL
DATAACCESS	DSNADM	*db2-subsystem*.DATAACCESS
EXPLAIN	DSNADM	*db2-subsystem*.EXPLAIN
SECADM	DSNADM	*db2-subsystem*.SECADM
SQLADM	MDSNSM	*db2-subsystem*.SQLADM
System DBADM	DSNADM	*db2-subsystem*.SYSDBADM

DB2 V10: Other New Security Functions

- **Separation of Duties**
  - You can configure DB2 to prevent users with SYSADM authority from altering authorizations, thus restricting security-related work to SECADM users.
  - This is done by setting the “SEPARATE SECURITY” ZPARM to ‘YES’
  - When SEPARATE_SECURITY is set to ‘YES’, then the SYSADM and SYSCTRL authorities cannot be used to affect the security characteristics of the system. Specifically:
    - The SYSADM authority does not allow the management of security objects, such as roles and trusted contexts.
    - The SYSCTRL authority does not allow the management of roles.
    - The SYSADM and SYSCTRL authorities cannot perform grants and cannot revoke privileges granted by others.
- **Row and Column Access**
  - DB2 allows you to restrict access to the contents of a table by row by and column
- **Significant logging enhancements**
- **For more detail, see session RAA12, “DB2 V10 Security Features – A New Standard in Data Protection”, Gayathiri Chandran, Thursday, 10:45 AM.**

z/OS V1.13 Preview Announcement for RACF

z/OS V1.13 Preview Announcment (RACF)

- **RACF Remote Sharing Facility (RRSF)** will be designed to **support the use of TCP/IP connections**, in addition to the current support for SNA Advanced Peer-to-Peer Communications(APPC). When used with TCP/IP, **RRSF will be designed to use Application-Transparent Transport Layer Security (AT-TS) to authenticate peer RRSF nodes and encrypt replication traffic.** AT-TLS provides encryption algorithms **thought to be stronger than those available using APPC**. A sample rule that specifies the strongest available encryption method is planned to be provided. The use of TCP/IP is intended to help improve usability, simplify network configuration, and improve the security of RACF data shared between RACF nodes in the RRSF network.

z/OS V1.13 Preview Announcement (RACF)…

- **What this means is that you can:**
  - Manage your RRSF network using the same skills as the rest of your TCP/IP network
  - Ensure that the same network security policy (IDS, IPS, etc.) is in place for your RRSF network as in place for the rest of your z/OS TCP/IP network
  - Utilize the encryption and peer-node authentication of AT-TLS
  - Keep up with improvements in z/OS Communications Server Security

z/OS V1.13 Preview Announcement (RACF)…

- **RACF is planned to support hardware-generated Elliptic Curve Cryptography (ECC) secure keys**, giving you the ability to issue and use certificates hardware-protected ECC keys**.**

- **RACF support is planned for generating Elliptic Curve Cryptography (ECC) secure keys using the Crypto Express3 Cryptographic Coprocessors (CEX3C) available for zEnterprise servers.** New keywords on the RACDCERT command are designed to allow **you to specify that an ECC key be stored in the ICSF public key data set (PKDS). Corresponding hardware ECC key support is planned for PKI Services.** This new support is intended to allow you to expand your use of certificates with ECC keys protected by hardware.

z/OS V1.13 Preview Announcement (PKI)…

- **For z/OS V1.13, z/OS PKI Services** is planned to **add support for DB2 9 for z/OS or later as its back-end key store**, enabling enterprise-class scale and resilient certificate management.
- **What this means to you:**
  - Leveraging DB2's capabilities for storage and retrieval of large numbers of digital certificates
  - Remove the 32K limit on Certificate Revocation Lists (CRLs)

z/OS V1.13 Statement of Direction (RACF)

- **z/OS V1.13 is planned to be the last release to support BPX.DEFAULT.USER. IBM** recommends that you **either use the BPX.UNIQUE.USER support that was introduced in z/OS V1.11, or assign unique UIDs to users who need them and assign GIDs for their groups.**

z/OS V1.13 Statements of Direction (RACF)…

- **Background: Assigning UID and GIDs**
  - **RACF 2.1 (1994):** Introduced OMVS segments for USERs and GROUPs.
    - Users with an OMVS segment could now use “Open MVS” (now z/OS UNIX System Services)
  - **OS/390 R2.4 (1997**): Introduced BPX.DEFAULT.USER FACILITY class profile
    - Allows assigning UIDs and GIDs to users and groups who do not have OMVS segments; ***One UID and one GID for all default users***

z/OS V1.13 Statements of Direction (RACF)…

- **Background: Assigning UID and GIDs…**
  - **z/OS V1.4 (2002):** Introduced AUTOUID/AUTOGID keyword on ADDUSER, ALTUSER, ADDGROUP, ALTGROUP
    - RACF could now find the next available UID or GID using the BPX.NEXT.USER profile in the FACILITY class
    - Required enabling RACF Alternate Index Mapping (“AIM”) to stage 2
      - Limitation of 129 eight-character users sharing one UID
      - Required running migration utility (“IRRIRA00”)
  - **z/OS V1.11 (2009):** Automatic generation of OMVS segment for USERs and groups
    - Built upon AUTOUID/AUTOGID
    - Requires AIM stage 3
    - Uses the BPX.UNIQUE.PROFILE in the FACILITY class

z/OS V1.13 Statements of Direction (RACF)…

- **What this means to you:**
  1. If you are using BPX.UNIQUE.USER then:
     - You are not using BPX.DEFAULT.USER (even if it is defined)
     - This SoD has no impact to you.
  2. If you are already assigning UIDs and GIDs to all users using z/OS UNIX System Services by assigning OMVS segments to all necessary users and groups, then:
     - You must continue to assign all new users and groups OMVS segments
  3. If you are already assigning UIDs and GIDS to all users user z/OS UNIX Sstem Services by defining OMVS segments using AUTOUID/AUTOGID (which uses BPX.NEXT.USER) then:
     - You are already using AIM at a minimum of stage 2
     - You must continue to assign all new users and groups OMVS segments
  4. If you are using only BPX.DEFAULT.USER
     - You must either move to the automatic generation of OMVS user and group segments or assign OMVS user and group segments to all necessary users and groups

Appendix
z/OS V1.11

z/OS V1.11: Program Object Signature Verification

- **Allows the signing of program objects and the verification of the signature of program objects when the objects are loaded into storage**
  - BINDER: Creates signatures by calling RACF when the SIGN option has been specified
  - RACF: Stores the information (certificates, keys, and options) necessary for the signature generation and validation, calculates the signatures, performs the validations, and logs the results.
  - LOADER: Calls RACF when program objects are loaded
- **You can sign your own code and vendors can sign theirs**

z/OS V1.11: Program Object Signature Verification…

- **Why sign code?**
  - “Belts and suspenders” or “defense in depth”: This support is intended to be used in conjunction with existing security mechanisms .
  - Digitally signing code can help increase the reliability and security of the system by adding an additional layer of controls on executable programs running on the system.
    - Digitally signing code makes it possible to detect changes to programs due to tampering or corruption.
    - Requiring that certain code be signed makes it easier to enforce change control procedures and protect against accidental changes to program code libraries. This helps avoid errors such as accidently placing 'test' code on a 'production' system.

z/OS V1.11: Program Object Signature Verification…

- **RACF profiles are used to control program signing:**
  - Key ring associated with the user performing the signing
    - Contains the information appropriate for program signing (private key, X.509 certificates (signing, CertAuth) which themselves must be appropriately signed
  - IRR.PROGRAM.SIGNING profile(s) in the FACILITY class
    - Used to associate the key ring owner, key ring name, and message digest algorithm used in the signature generation and validation process.

z/OS V1.11: Program Object Signature Verification…

- **RACF profiles are used to control program verification:**
  - IRR.PROGRAM.SIGNATURE.VERIFICATION profile in the FACILITY class
    - Used to associate the key ring owner and key ring name of the key ring which contains the signature verification key ring
  - Profiles in the PROGRAM class
    - Contains information options that specify the actions to be taken during verification process:
      - SIGREQUIRED: Is a signature required for this program? (YES,NO)
      - FAILLOAD: Under what conditions should the load fail? (ANYBAD, BADSIGONLY, NEVER)
      - SIGAUDIT: What should be logged? (ALL, SUCCESS, ANYBAD, BADSIGONLY, NONE)

z/OS V1.11: Program Object Signature Verification…

- **Considerations:**
  - Only program objects (which must reside in in PDSEs) can be signed and verified.
    - Code in PDS or z/OS Unix System Services file system, or non Program Object code cannot be signed and verified. However, z/OS UNIX programs can 'link' to signed executables in PDSEs.
  - If a signed program is zapped (executable code changed), its signature is no longer valid.
  - IBM ships portions of the System SSL product as signed code.
  - Support is new for z/OS R11 and has been rolled back to z/OS R10.
  - Any installation or software provider can use these services to sign their own code.
  - Program objects are not encrypted

z/OS V1.11: Logon Statistics Suppression

- **Allows you to specify which applications should only record on the first system access of a day**
  - Why? Reduced I/O and lower the impact of serialization on the RACF dataset.
- **APPL profiles are used to specify which applications are taking advantage of logon statistics suppression**
  - Specify “RACF-INITSTATS(DAILY)” anywhere in the APPLDATA
  - APPL class must be active and RACLISTed

z/OS V1.11: Identity Propagation

- **Prior to z/OS V1.11, clients using distributed server applications which used a common server or application identity for transaction executing on z/OS would not be able to pass the identity of the end user to z/OS for logging**
- **With z/OS V1.11, applications can pass the distributed identity information abut the end user (distinguished name and realm) into z/OS where it will be used for logging**
  - Exploited by CICS TS 4.1
- **The distributed identity can be mapped to a RACF identity at:**
  - the distributed application server (as is often done today) or
  - the execution point on z/OS, using the new RACMAP support

RACMAP [ID(*mapped-to-userID***)]**
MAP
USERDIDFILTER(NAME('*distributed-identity-username-filter***'))**
REGISTRY(NAME('*distributed-identity-registryname***'))**
[WITHLABEL('*label-name***')]**
| DELMAP[(LABEL('*label-name***'))]**
| LISTMAP[(LABEL('*label-name***'))]**

z/OS V1.11: R_admin Enhancements

- **R_admin can now be used to extract information about general resources**
  - Extract specified profile - ADMN_XTR_RESOURCE (X'1F')
  - Extract next profile - ADMN_XTR_NEXT_RESOURCE (X'20')
- **Authorization required for problem state callers:**
  - At least READ access to the IRR.RADMIN.RLIST resource in the FACILITY class
  - Users are limited to seeing only the information that would be displayed by an RLIST command
    - For example , audit settings will be suppressed if caller does not have the AUDITOR attribute
- **Supervisor callers can request either, both, or no check**
  - Command authority enforced by default

z/OS V1.11: R_admin Enhancements…

- **R_admin SETROPTS option extraction (ADMN_XTR_SETR (X'16')) may now be called from problem state**
- **Authorization required for problem state caller:**
  - At least READ access to IRR.RADMIN.SETROPTS.LIST in the FACILITY class
  - Authority as enforced by the SETROPTS command
    - For example, audit settings will be suppressed if caller does not have the AUDITOR attribute
- **No changes required to existing programs other than to remove MODESET into supervisor state**

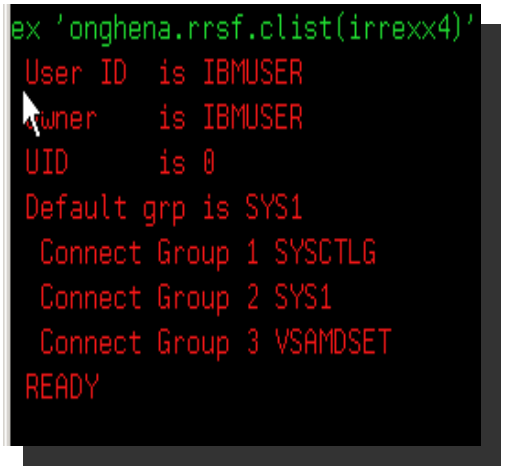
z/OS V1.11: LDAP Change Logging of General Resources

- **You can now tell RACF to create change log entries for changes to general resources by defining the profile NOTIFY.LDAP.*class-name* in the RACFEVNT class and activate the class**
- **Events which are logged:**
  - Resource additions made using the **RDEFINE** command
  - Resource modifications made using the **RALTER** command
  - Changes to the resource's access list using the **PERMIT** command
  - Resource deletions made using the **RDELETE** command
- **ICHEINTY/RACROUTE applications can create their own change log entries using R_proxyserv (IRRSPY00)**

z/OS V1.11: REXX Interface to R_admin Extract Functions

- **IRRXUTIL allows you to extract information from the RACF database using the REXX programming language**
- **Data is returned as stem variables**

```
/* REXX */
myrc=IRRXUTIL("EXTRACT","USER","IBMUSER","RACF")
  if (word(myrc,1)=0) then do
    say "User ID  is "RACF.PROFILE
    say "Owner    is "RACF.BASE.OWNER.1
    say "UID      is "RACF.OMVS.UID.1
    say "Default grp is "RACF.BASE.DFLTGRP.1
    do i=1 to RACF.BASE.CGROUP.0
       say " Connect Group "i" "RACF.BASE.CGROUP.i
    end
  end
```



z/OS V1.11: Automatic UID/GID Assignment

- **z/OS UNIX System Services tasks are associated with user and group identifiers (UIDs & GIDs)**
  - Can be assigned explicitly in RACF profiles (preferred)
    - AUTOUID/AUTOGID can be specified to generate a unique UID/GID
  - Can default from BPX.DEFAULT.USER profile
- **New option to assign permanent unique UIDs/GIDs is enabled by BPX.UNIQUE.USER profile in FACILITY class. Once enabled, RACF and UNIX System Services:**
  - Create a unique UID/GID and
  - Generates an OMVS segment for the user/group if none exists
    - APPLDATA specifies a default user profile from which the other segment information is copied.
  - Uses the existing BPX.NEXT.USER processing (from AUTOUID/AUTOGID)
- **Implemented in initUSP, getUMAP, & getGMAP, which are invoked by various UNIX system services**

z/OS V1.11: Profile Name in Authorization Exits

- **The RACROUTE REQUEST=AUTH (ICHRCX02) and REQUEST=FASTAUTH(ICHRFX02,ICHRFX04) exits have always received a pointer to the profile which was used in the access control decision**
  - Profile is one which allowed or denied the request
  - Can differ from the resource name (if a generic profile was matched)
- **With z/OS V1.11, the exits receive the name of the profile as well**
  - For REQUEST=FASTAUTH, if the profile name is generic, then the internal format of the profile name is returned
  - RACROUTE REQUEST=AUTH, the profile name is always in external format
  - A new service is provided to map the internal format of the profile name to the external format

z/OS V1.11: RACF Enhancements for ICSF

- **New ICSF segment on CSFKEYS, GCSFKEYS, XCSFKEY, and GXCSFKEY profiles allows the specification of controls on high performance secure keys**
  - **ASYMUSAGE([NO]SECUREEXPORT [NO]HANDSHAKE):** Defines allowable usages of the asymmetric key(s) covered by this profile
  - **SYMEXPORTABLE(BYANY | BYLIST | BYNONE | NOSYNEXPORTABLE)**: When can the symmetric be exported? When BYLIST is chosen, the key can only be exported by an asymmetric key contained in one of the following lists.
  - **SYMEXPORTCERTS(cert-label1, cert-label2, …):** Identifies the certificates whose keys may be used to export the symmetric key. These certificates must exist within the certificate container (SAF key ring or PKCS#11 token) defined to ICSF via an ICSF setting. “*” means any certificate in the container may be used.
  - **SYMEXPORTKEYS(key-label1, key-label2, …):** Identifies the PKDS labels of keys which may be used to export the symmetric key. “*” means any key in the PKDS may be used.

z/OS V1.11: RACF Enhancements for ICSF…

- **ICSF segment fields may be extracted using RACROUTE REQUEST=EXTRACT,BRANCH=YES**
- **Mapping of in-storage ICSF segment information is in ICHPISP SAF mapping macro**
  - ICSF segment information is available to REQUEST=FASTAUTH for locally RACLISTed profiles and to REQUEST=FASTAUTH exits
- **ICSF segment is unloaded by IRRDBU00 as record type 05G0**
- **RACF panels are populated with initial values for the ICSF segment**
  - … as long as the user is authorized to list the fields and is allowed to use the R_admin interface, which requires READ authority to IRR.RADMIN.RLIST in the facility class

z/OS V1.11: SMF Unload of WAS & TKLM SMF data

- **Support for z/OS SMF repository of audit data**
  - WebSphere Application Server (WAS) V7
  - Tivoli Key Lifecycle Manager (TKLM)
- **Both WAS and TKLM create SMF Type 83 records via the z/OS JAVA class interface to the R_auditx service**
  - Subtype 5: WebSphere
  - Subtype 6: TKLM
- **IRRADU00 unloads XML and traditional tabular-format output files of audit data**
- **LRECL of tabular IRRADU00 output is now 12288**
  - Will be adjusted automatically by IRRADU00

z/OS V1.11: Digital Certificate Support

- **RACDCERT multi-byte character improvements**
  - Support (installation, retrieval and authentication) for certificates which contain characters which are outside the 1047 code page.
  - If a character does not map to code page 1047, the character will be represented by 6 characters in the format of U+nnnn, where nnnn is the Unicode code point of that character in hexadecimal format
  - When the certificate profile is created, the 6-character format will contribute to the profile name.
    - There is a risk of exceeding the profile name limit, which will prevent the creation of the certificate in RACF.
- **PKI Private Key recovery**
  - Prior to z/OS V1.11, PKI services did not generate private/public key pairs. In R11, key generation and key archival capabilities are being introduced. The certificate requestor will have the option to generate the public/private key pair themselves as in previous releases or have PKI Services generate the key pair.

z/OS V1.11: Digital Certificate Support…

- **PKI Web Pages**
  - PKI services now provides Java server pages (JSPs) and an XML template file to create and customize the PKI Services Web application as an alternative to the existing REXX CGI support.
- **PKI Support for SHA256 with RSA signature algorithm**
  - PKI Services will support the "SHA256 with RSA encryption" signature algorithm for signing certificates, certificate and authority revocation lists (CRL/ARL), and OCSP responses