CICS Security – The Basics

Leanne Wilson

April 2025

Agenda

- Introduction
- Objectives
- CICS Security
- Summary

Introduction

Leanne Wilson

Senior Security Consultant/ Technical Delivery Manager

Worked on mainframes for 13 years

Worked on many security-based projects

Also worked as an Information Security Manager concerned with GRC.

VERTALI

In My Spare Time

OBJECTIVES

Objectives

- Learn the key security components of CICS
- Explore authentication, authorization, and auditing mechanisms
- Identify best practices for securing CICS environments

- This session will delve a little into CICS security and give you, hopefully enough information to go and understand your own CICS implementation from a security perspective

CICS & RACF BASICS

What is CICS?

- **C**ustomer **I**nformation **C**ontrol **S**ystem (CICS)
- A transaction processing system, that for some reason has become quite popular over the years, who's role is to provide online transaction processing (OLTP)
- Not the only transaction processing subsystem that IBM has:
  - IMS and Websphere and there are others
- Specialised infrastructure that supports multiple users and processes multiple application programs concurrently

Language

CICS System

CICS Started Task

CICS Region

CICSPLEX – clustered version of the above

What is CICS?

- Multi-Region Operation (MRO) - within one z/OS system or Sysplex
- Single instance of CICS

- Provides interface to other systems - DB2, IMS, IDMS

- First commercial release July 8th 1969

RACF

Resource	Resource Access Control Facility
Identify and authenticate	Identify and authenticate users
Authorise	Authorise users to access protected resources
Log and report	Log and report attempted unauthorised access
Control	Control means of access to resources

Basic RACF Example

The group he CEMT transa

In CICS te

In RACF, a

- RDEFI
- PERM

VERTALI

CICS SECURITY

The CSD (CICS System Definition)

- CICS System Definition (CSD) is a VSAM dataset where resource definitions are stored
- Access to this file and the transactions and batch utilities that manipulate and list its contents need to be strictly controlled
- The CSD is updated using transactions CEDA and CEDB
- The CSD is viewed using CEDC
- DFHCSDUP a CICS supplied batch utility can be used to list the contents of the CSD

The SIT!

- Defines configuration options for a CICS region
- SIT parameters govern the RACF interface

VERTALI

Systems Initialisation Table(SIT)

- Parameter settings obtained from:
  - Built-in CICS defaults
  - DFHSITxx Macro assembly modules (default DFHSIT)
  - SYSIN DD Statement
  - EXEC Statement PARM
  - Console commands; However you cannot change security parameters via the console

- Last parameter definition found is the one used

Example SIT

```
SIT TITLE 'DFHSIT - CICS DEFAULT SYSTEM INITIALIZATION TABLE'
    DFHSIT TYPE=CSECT,
        APPLID=VRTCICS,                 VTAM APPL identifier
        CMDSEC=ASIS,                    API command security checking
        DFLTUSER=CICSUSER,              Default user
        PLTPISEC=NONE,                  No PLT security checks on PI programs
        PLTPIUSR=,                      PLT PI userid = CICS region userid
        SEC=YES,                        External security manager option
        SECPRFX=NO,                     Security prefix
        USRDELAY=30                     Delay before ACEE refresh
        XCMD=YES,                       Use default RACF class name
        XDCT=NO,                        Do not perform RACF check
        XFCT=$UKFCT,                    FCT use UK class for RACF check
        XJCT=NO,                        Do not perform RACF check
        XPCT=YES,                       Use default RACF class name
        XPPT=YES,                       Use default RACF class name
        XPSB=YES,                       Use default RACF class name
        XTRAN=YES,                      Use default RACF class name
        XUSER=YES                       Surrogate user checking to be done
```


VERTALI

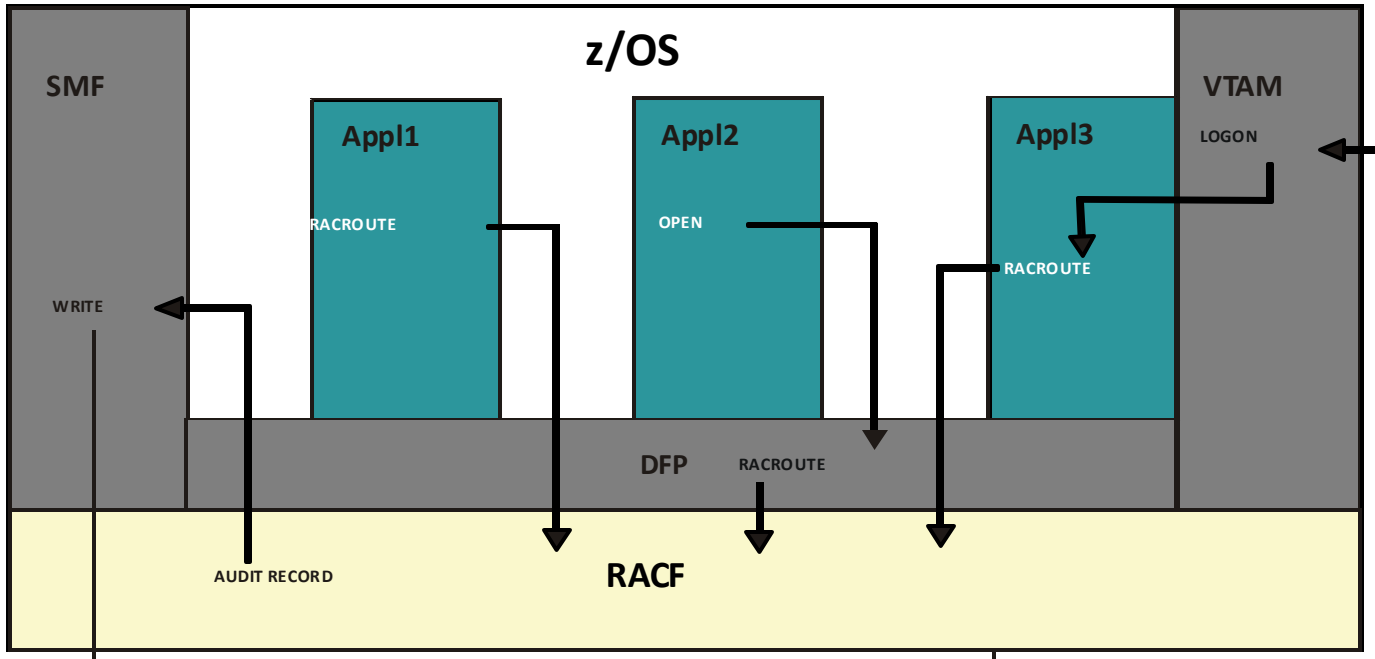
What the ESM does for us?

- CICS relies on an ESM (RACF, ACF2 or TSS) to provide security
  - The ESM controls
    - Who can logon to CICS
    - Who can execute a transaction
    - Who can use a transaction resource
      - Started Transaction
      - Program, File or Journal
      - Transient Data Destination
      - Temporary Storage Queue
    - Who can execute a CICS command (CMDSEC)

CICS and External Security

- CICS when configured to do so will call an External Security Manager (ESM)
- The ESM can be RACF, CA-ACF/2 or CA-Top Secret
- CICS has no mechanism today for internal security
- If you don’t use an ESM, you have NO security!

The CICS RACF Interface



Profiles

- User
- Group
- Dataset
- Resources

SMF Data Sets

Audit Reports

RACF Database

VERTALI

The role of CICS in security control

- To invoke SAF via RACROUTE to perform:
  - User Signon/Signoff
  - Access Authorisation

SIT!

- The SIT as previously mentioned is where most of the good stuff happens!
- We must understand this in detail and all the parameters that are here!
- You need to strictly control the SIT
- Some environments control it using Endevor, ISPW or Change man

VERTALI

SIT Parms

```
APPLID=(VRTCICS),
FCT=NO,
GMTEXT='Vertali CICS SYSTEM',
GRPLIST=(XYZLIST,CICSTS32),
IRCSTRT=YES,
ISC=YES,
STATRCD=ON,
SEC=NO,
TCT=NO,
TRTABSZ=64,
XRF=NO
```

So what's the problem with these parameters?

Hint: Something to do with Security

VERTALI

Is security being used?

- SEC=NO
  - No External Security Manager being used

- SEC=YES
  - External Security Manager is being used

What is being protected?

- Controlled by several other parameters in sit in the form Xnnnn=

- Where:
  - XTRAN = Transaction Security
  - XFCT = File Control Security
  - XCMD = Command Security
  - XTST = CICS Temp. storage control
  - XPCT = Started Transaction control

  - To name but a few!

Xnnn SIT Parameters

- The majority of the Xnnn SIT parameters follow the form:
  - **NO**
    - Option is disabled
  - **YES**
    - Option is enabled with default IBM RACF classes
  - **Class_name**
    - The installation has created a site specific RACF class, normally a pair member & grouping

What controls are available to protect the data?

Transaction security - XTRAN
Always configure CICS regions to use transaction security. However, transaction security is in effect a form of boundary checking. Therefore, it's not sufficient for a zero trust strategy on its own. Unless additional controls are applied, a transaction can access any resource in the CICS region.

Resource security - RESSEC
Transaction security is not sufficient for a zero trust strategy on its own. It is recommended that you use resource security to protect data that is accessed by an application for an additional layer of security. Enable resource security for all classes of resources that you need to

protect

Command security - CMDSEC
System programming interface (SPI) commands can be used to manage the CICS system and its resources. It is recommended that you use command security to control who is allowed to issue the commands.

CICS TRANSACTION SECURITY

XTRAN SIT Parameter

- If XTRAN=YES
  - then the IBM supplied RACF classes TCICSTRN & GCICSTRN are being used for transaction security

- If XTRAN=£PRDTRN
  - then the site defined RACF classes T£PRDTRN &

    G£PRDTRN RACF classes are being used for transaction security
  - Note that CICS enforces the first character of the RACF class for transaction profiles to be a T
  - Other resource types have their own rules

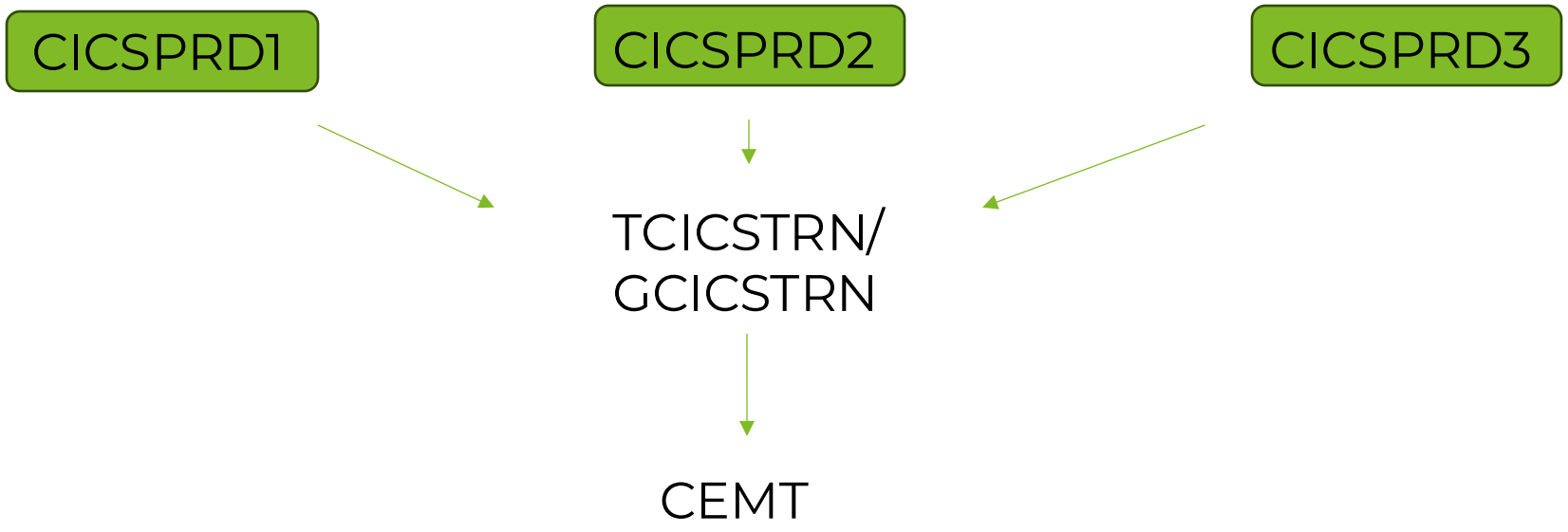
Transaction Profiles and Classes

- You have some choices….

- Do I want to use the standard IBM classes TCICSTRN and GCICSTRN?
- Do I want to create my own classes?
- Do I want the profiles prefixed?

RACF Classes and Prefixing 1

- Share default classes among CICS regions
  - TCICSTRN and GCICSTRN shared between CICSPRD1, CICSPRD2 and CICSPRD3
  - Would all have XTRAN=YES defined in their respective SIT's
  - SECPRFX=NO
  - They could even be sharing the same SIT parameter dataset



RACF Classes and Prefixing 2

- Create locally defined independent classes for each region or set of related regions (Prod, Dev, Test, etc)

  - Eg T£PRDTRN and G£PRDTRN shared between CICSPRD1, CICSPRD2 and CICSPRD3
  - Eg T£DEVTRN and G£DEVTRN shared between CICSDEV1, CICSDEV2 and CICSDEV3
  - Would all have XTRAN=£PRDTRN / XTRAN=£DEVTRN set in their SIT
  - SECPRFX=NO

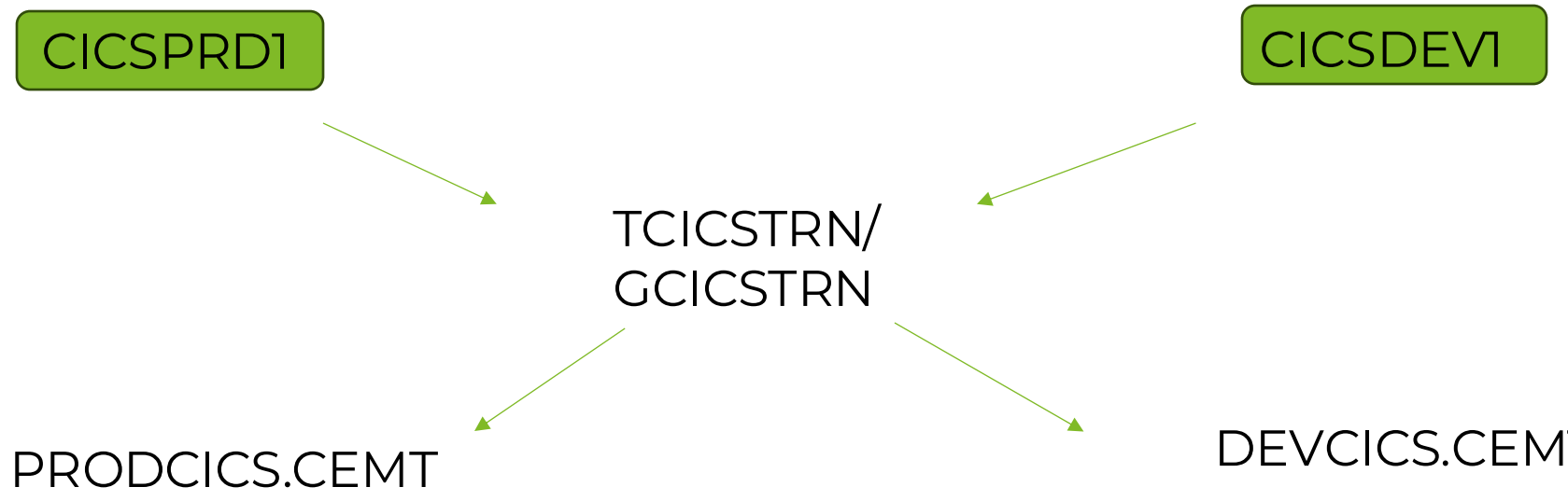
RACF Classes and Prefixing 3

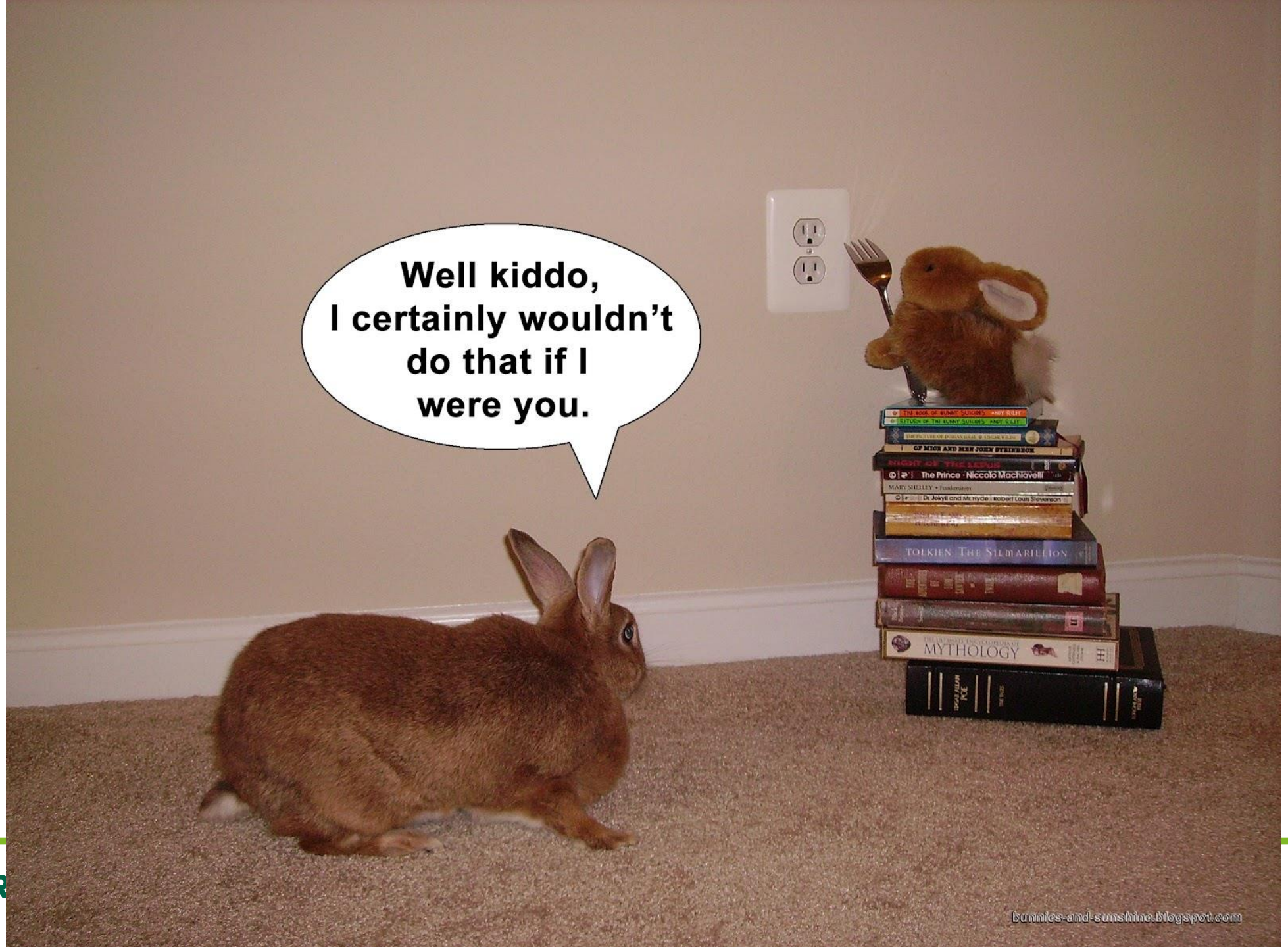
- Classes shared by dissimilar CICS regions
  - May need to differentiate resources belonging to specific CICS regions
  - Resource names can be prefixed with CICS region's USERID or Prefix
  - SIT Parameter - SECPRFX=YES | **NO** | *Prefix*

- CEMT in Prod needs to be locked down; but available in Dev
  - Prod userid is **PRODCICS** and Dev is **DEVCICS**
  - Two RACF profiles **PRODCICS.CEMT** and **DEVCICS.CEMT**

RACF Classes and Prefixing 3

- TCICSTRN and GCICSTRN shared between CICSPRD1 and CICSDEV1
- Would have XTRAN=YES defined in their respective SIT's
- SECPRFX=YES




Well kiddo,
I certainly wouldn't
do that if I
were you.
The Prince · Niccolò Machiavelli
TOLKIEN THE SILMARILLION
MYTHOLOGY
bunnies-and-sunshine.blogspot.com

My Recommendation

- I would always go with separate RACF classes and not use prefixing
- It's easy now that we have the RACF CDT class
- But ultimately, you must do what is best for your organisation

What RACF profiles do I have?

- Use the RACF SEARCH command to list all of the profiles in a given class:
  - SR CLASS(TCICSTRN) NOMASK
  - SR CLASS(GCICSTRN) NOMASK
- This only shows the profiles

Who has access to them?

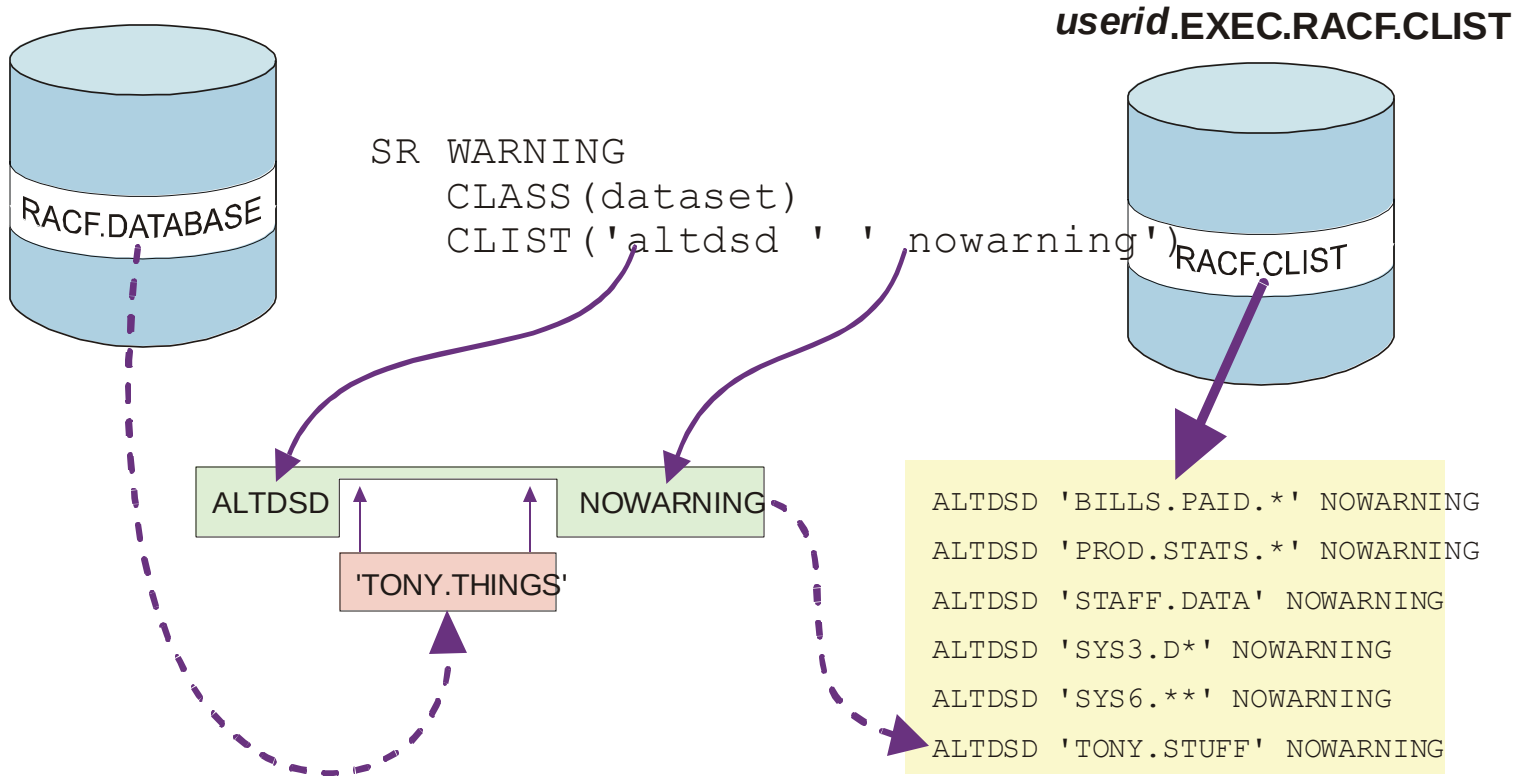
- You need to list each profile and check:
  - UACC
  - Access List
  - Conditional Access List

- You can generate the required RACF commands with the SEARCH command and CLIST option

Search Command – CLIST Option

SEARCH xxx. xxx

CLIST ('string1' 'string2')



VERTALI

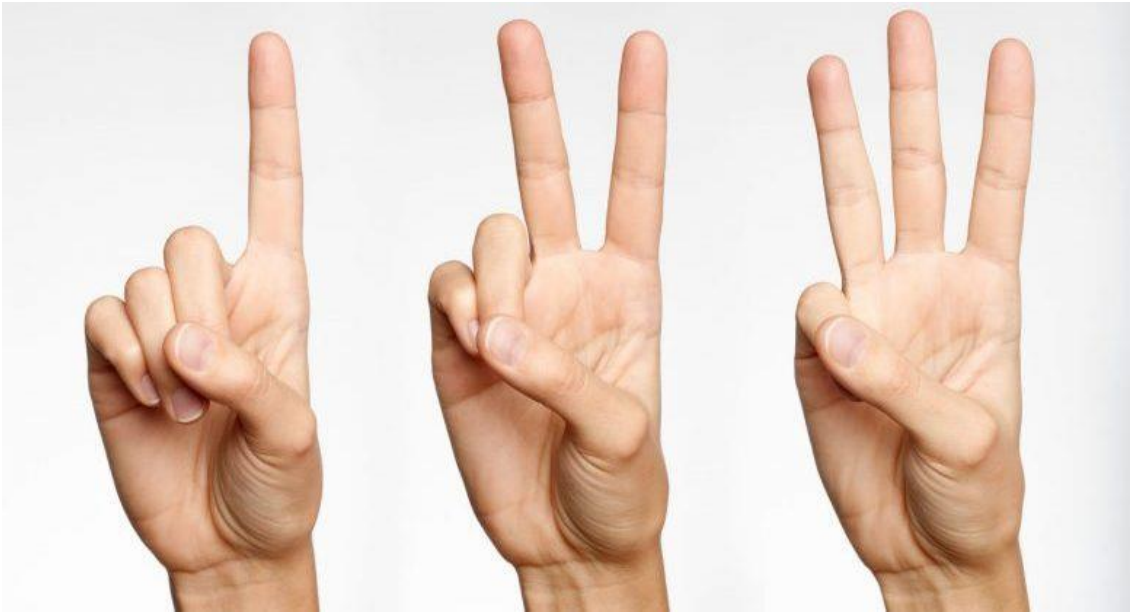
Example Search in batch

```
//SEARCH EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
 SR CLASS(TCICSTRN) NOMASK NOLIST -
  CLIST('RL TCICSTRN ' ' ALL')
//*
//EXEC   EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
 EX EXEC.RACF.CLIST
```

CICS Transaction Security

- IBM supply many transactions as part of the basic CICS install

- They are categorised and all have different security requirements



Category 1

- CICS Internal use only
- Never associated with a terminal
- RACF (ESM) is NOT called for these transactions
- Some people define them to RACF for documentation purposes

Category 2

- CICS Administration transactions
- Very powerful
- Very restricted access lists
- All RACF profiles should have a UACC of NONE
- May be a good candidate for AUDIT(ALL(READ)) to log all access successful or not
- Check the manuals carefully there are additional security requirements/suggestions

Category 3

- All users require access to these transactions
- All Category 3 transactions are exempt from security checks
- Some people define them to RACF for documentation purposes

VERTALI

MEMBER AND GROUPING CLASSES

A PAIR OF CLASSES

- **Member Class and a Grouping Class**

RACF CLASS NAMES

Member class	Resource grouping class	Description
TCICSTRN	GCICSTRN	CICS transactions, normal attach security
PCICSPSB	QCICSPSB	CICS PSBs
ACICSPCT	BCICSPCT	CICS-started transactions
DCICSDCT	ECICSDCT	CICS transient data queues
FCICSFCT	HCICSFCT	CICS files
JCICSJCT	KCICSJCT	CICS journals
MCICSPPT	NCICSPPT	CICS programs
SCICSTST	UCICSTST	CICS temporary storage queues
CCICSCMD	VCICSCMD	EXEC CICS SYSTEM commands
RCICSRES	WCICSRES	Document templates, bundles, EP adapters, EP adapter sets, event bindings, ATOMSERVICE definitions, and XML transforms

Member or Grouping Class?

- What are they?
- Two different ways to protect resources in CICS
- How does CICS use them?
  - Profile merge
  - In Storage profiles
- Who has access?

Example of Member Class Profiles

- The warehouse group of users need access to three transactions: INVC, ORDP & STOH

```
RDEFINE  TCICSTRN  INVC  OWNER(SECADM)  UACC(NONE)
RDEFINE  TCICSTRN  ORDP  OWNER(SECADM)  UACC(NONE)
RDEFINE  TCICSTRN  STOH  OWNER(SECADM)  UACC(NONE)
PERMIT  INVC  CLASS(TCICSTRN)  ID(WHSEUSRS)  ACCESS(READ)
PERMIT  ORDP  CLASS(TCICSTRN)  ID(WHSEUSRS)  ACCESS(READ)
PERMIT  STOH  CLASS(TCICSTRN)  ID(WHSEUSRS)  ACCESS(READ)
```

Example Grouping Class Profiles

- The warehouse group of users need access to three transactions: INVC, ORDP & STOH

```
RDEFINE  GCICSTRN  WARE_TRNS  OWNER(SECADM)  UACC(NONE)
RALTER  GCICSTRN  WARE_TRNS  ADDMEM(INVC  ORDP  STOH)

PERMIT WARE_TRNS CLASS(GCICSTRN) ID(WHSEUSRS) ACCESS(READ)
```

How RACF merges Profiles

- Member / grouping classes must be loaded into memory
- Applies only to member / grouping classes
- Merge applies only if a resource name appears in more than one profile
- UACC: The most restrictive UACC is chosen from the profiles that are merged
- Access list: If a user or group appears in the access lists of multiple profiles, that user or group is given the highest access

OTHER BITS

User Logon at Terminal

- RACF logon
  - CESN sign-on transaction
  - Program with EXEC CICS SIGNON command
  - CICS Supports MFA, just saying

- APPL applid - as determine by SIT parameters
  - APPLID= Region’s application ID
  - READ access required

User Logon at Terminal

- CICS concurrent logon restrictions
  - SNSCOPE=NONE | CICS | MVSIMAGE | SYSPLEX
    - NONE — No restriction
    - CICS — Only once in each CICS region
    - MVSIMAGE — Only once for entire MVS image
    - SYSPLEX — Only once for entire Sysplex
- Only effects user logon via CESN
- Does not affect pre-set terminal logons

Auditing CICS

- SMF 1154 subtype 80 records from CICS
- IBM Healthchecker
- Review CICS job logs
- Issued SPI Commands
- Auditing RACF CICS resources
  - Profiles
  - ACLS
  - Audit settings

Thank You

VERTALI

VERTALI