SHARE Winter 2026 – Orlando, FL

CICS TS 6 – Journey to Zero Trust

Lewis James

CICS TS for z/OS Development

IBM UK

IBM X-Force Research 2025

SHARE
EDUCATE • NETWORK • INFLUENCE

30%

Abusing valid accounts remained the preferred entry point into victim environments for cybercriminals in 2024, representing 30% of all incidents X-Force responded to.

84%

Phishing emerged as a 'shadow' infection vector for identity attacks. While the share of successful phishing compromises has dropped by nearly 50% since 2022, X-Force observed an 84% uptick in phishing emails delivering infostealers on a weekly basis.

25%

Attackers exploited vulnerabilities in more than one-quarter of incidents X-Force responded to across critical sectors last year, with outdated systems and slow patching cycles proving to be an enduring challenge

When the auditor comes knocking…

Auditors inspecting security environments are looking at key compliance with industry standard regulations

EU

GDPR

DORA

NIS2

US

HIPAA

SOX

ISO

PCI-DSS

Complying with the regulations

It is easier said than done in ‘just complying’ with the regulations.

Large international corporations may be required to comply with multiple security regulations when

- The corporation has a presence within multiple geographies
- A national corp serving international customers handling and storing their PI

A common framework or strategy is required in centralizing the approach of cyber security

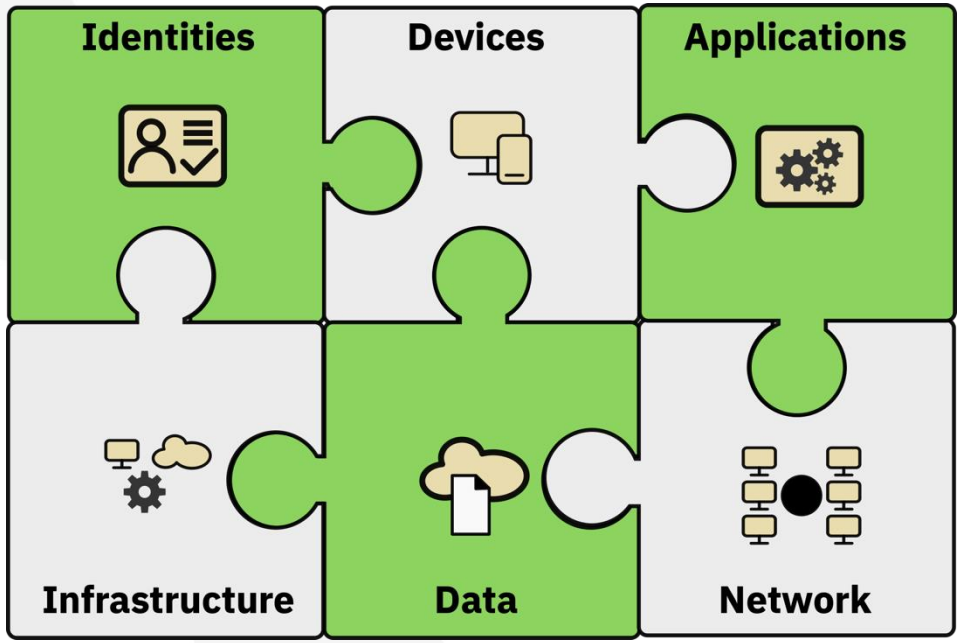
What is Zero Trust

Focus on protecting resources not perimeters

Enable the right user,
to have the right access,
to the right data,
for the right reasons.

Never trust, always verify

Assume the bad actor is already present and continuously monitor.



The Bad (*or good*) Actor

Traditionally enterprise security has focused around keeping the bad people out

Modern security frameworks – Zero Trust – looks at a different perspective

- What happens if your colleague is the bad actor?
- When a breach occurs how can we minimize the attack surface?
- How can we monitor if access is still required and make alterations to promote security?

Principle of Least Privilege (POLP)

Ensuring those who need that access only have that access - nothing more

Only granting minimum access required to perform the role.

No implied trust – always verify access.

How is it applied?

- Employing role-based security
- Effective monitoring and review
- Revoking user-granted permissions where possible

The Challenge

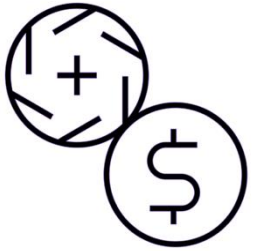
Migrating security environment to a Zero Trust model is not easy

Complex z/OS environments can often diminish feasibility

- Decades of security definitions built over one another
- Outdated accesses due to limited scope for monitoring
- No explicit IBM advice or tooling to promote good practice

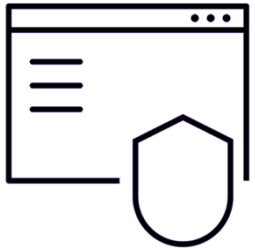
CICS Transaction Server for z/OS security aimed to fill this gap through version 6.

CICS Transaction Server for z/OS 6



Reduced cost of management and resiliency

CICS Admins can solve problems faster using OpenTelemetry observability, reduce costs by optimizing thread-safe access to data tables, reduce volumes of data written to SMF, and simplify and automate routine work with CICS configuration tools, CICS policies and Ansible

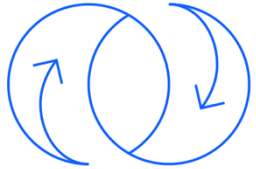


Improved security and compliance management

CICS and Security Admins can tighten security for valuable applications and data using CICS security recording and CICS Explorer tooling as part of a Zero Trust strategy, secure all connections with AT-TLS and TLS 1.3, and adopt best-practices with z/OS health checks

Enhanced developer productivity

Developers can use familiar tooling in Eclipse or VS Code to develop CICS applications. Java developers have access to the latest versions of Java, Jakarta, Spring Boot and Node.js to extend applications, with fast local access to CICS programs and data



Increased business agility

Architects can unlock access to CICS applications with API enablement, messaging event driven architecture, and AI in-transaction inferencing

OpenTelemetry

Manage and diagnose problems across hybrid cloud applications using OpenTelemetry dashboards and tools. CICS captures, propagates and emits trace spans for each task. Works across z/OS Connect, IBM MQ, CICS TG, Db2 and IMS for a complete end-to-end trace

Modernization

Develop CICS applications in VS Code with enhanced editors from IBM and Zowe. Developers can define and deploy CICS resources using YAML alongside the code, with auto completion and automation to ensure they meet local conventions

AI agent ready

Quickly answer questions about CICS regions and configuration using AI assistants in natural language. CICS includes an MCP server to provide accurate information to AI agents during app development and problem diagnosis

Core

Consolidate and simplify CICS region configuration using YAML for easier management and provisioning. New policies to issue messages to the system log and new policy rules for JVM servers and APPC connections

Security

Adopt Zero Trust principles and security best-practices with enhanced workflows in CICS Explorer, and and example security definition validation pipeline. AT-TLS can be used to secure IPIC and outbound HTTPS connections

Modern Languages

Enable developers to use the latest Java and Node.js support, including Java 21, Jakarta EE 10, and Spring Boot 3. CICS Java APIs have been further enhanced.

Theme of CICS Security

Over the last three releases (CICS TS 6)

Theme centered around simplification and compliance

Zero Trust at the forefront of all enhancements

- Different perspective of viewing security
- Understand and mitigate insider threats
- Assuming a bad actor is already present within the environment

Customer Research

CICS TS 6 security focusing on customer centric research and workshops

Most customers …

- Have not enabled resource security
- Do not test applications security until pre-production
- Do not have automation testing

Results from the CICS Futures Summit and Architecture Forum 2022

The Journey So Far

z/OS Security Compliance Centre – SMF 1154 Type 80 records

CICS Security Discovery

- Understanding the current security posture
- Analyzing the active security usage in production
- Building a more secure, role-based, environment

CICS Security Definition Capture - DevSecOps

- An an application level
  - Understand the security driven during an application
  - Build security metadata living with the source code
  - Provide a source of truth for the required access needed to run the application

CICS Compliance Evaluation

The current process

Auditor tries to interpret regulations for z/OS products

Auditor requests information based on their own interpretation

System programmer gathers SMF data, consoles, definitions, reports …

Auditor interprets that information

The problem

Auditor may have little understanding or experience with z/OS (or CICS)

Data collection is an expensive time-consuming process

Data collection should be a controlled and fair process

Interpretation is subjective, error-prone and often misunderstood

CICS Compliance Evaluation

The solution

Automate data collection as much as possible

Interpret and display compliance to specific regulations

Give trusted advice on compliance and best practice

Benefits

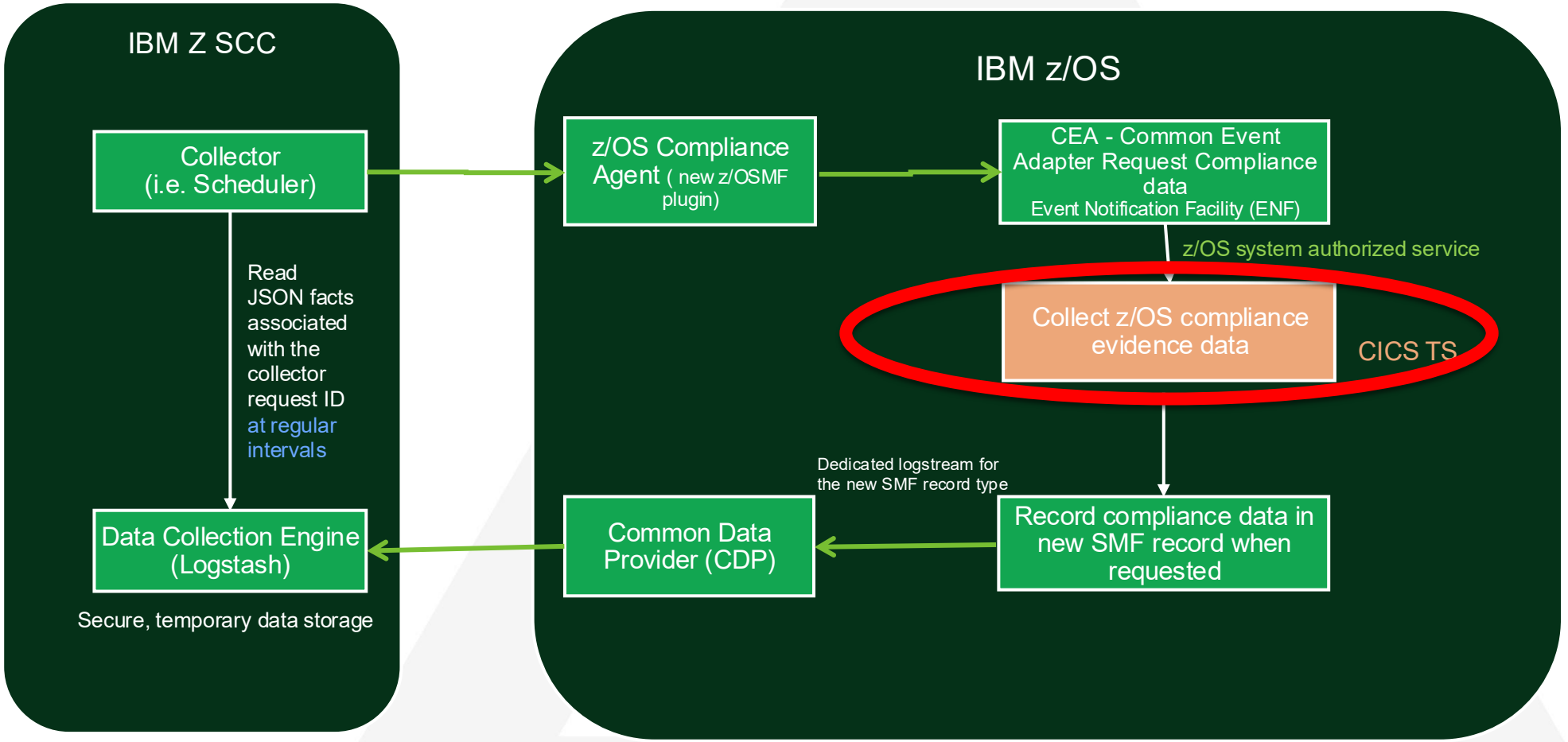
Auditor does not need to be a z/OS or CICS expert

Data collection is much cheaper

Collection process is fair and controlled

Consistent interpretation of security standards and regulations

z/OS Security Compliance Centre zSCC

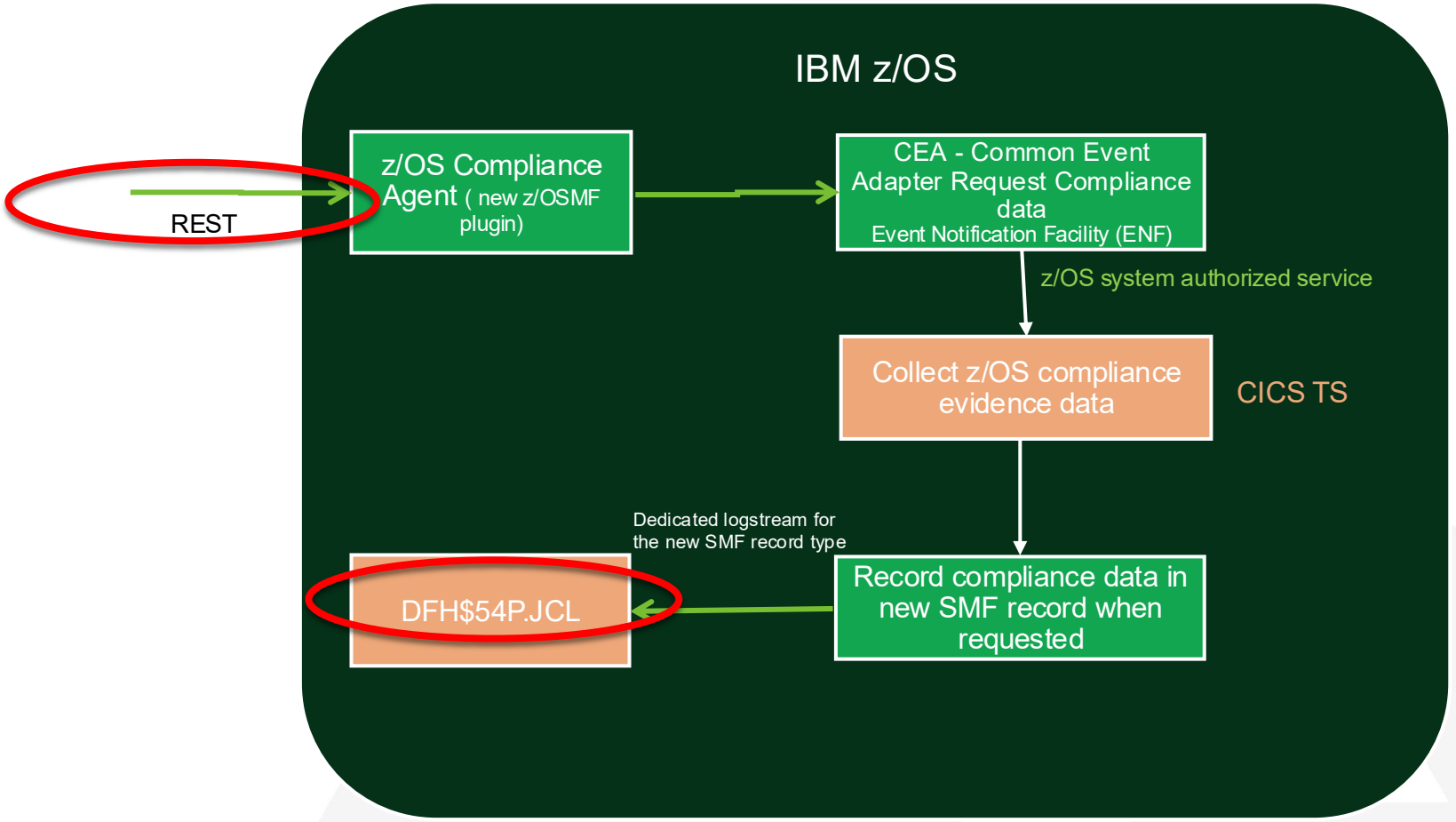


Current support
- PCI-DSS
- RACF
- Comms Server
- CICS TS

Requirements
- z/OS 2.4 +APARs
- IBM z16+
- For details see link

The z SCC requests compliance data from z/OS Compliance agent. The z/OS event notification facility (ENF) allows an authorized program to listen for the occurrence of a system event. When the ENF signal is received by a z/OS component, control passes to a listener user exit routine which would read current compliance evidence data and write it to a new SMF record type. The new SMF record would be filtered by CDP and transmitted to the z Compliance Authority. This is an asynchronous event.

CICS Compliance Data without zSCC



PCI-DSS

CICS

Requires z/OS 2.4 +APARs

- Not all customer will have the zSCC
- The RESTful interface is an authorized public interface (part of z/OSMF)
- CICS TS 6.1 provides a sample to print out SMF 1154-80 records

CICS Compliance Data - CSV

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	
RequestID	Sysplex	LPAR	Jobname	RegionID	APPLID	DFLTUSER	SEC	XUSER	RACFSYNC	CONFDATA	XCMD	XDB2	XDCT	XFCT	XJCT	XPCT	XPPT	XPSB	XRES	XTRAN	XTST	Regio
G6UPB2ELPLOW7UHP	PLEX2	MV2D	GB1220	D1	IYK4ZON1	CICSUSER	YES	YES	YES	SHOW	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
G6UPB2ELPLOW7UHP	PLEX2	MV2D	ALHUNTE	CMASOVC1	CAHCOVC1	ALHUNTE	NO	YES	YES	HIDE						CICSPCT			CICSRES			
G6UPB2ELPLOW7UHP	PLEX2	MV2D	PENFOLD	CICSDL3	IYK4ZDL3	CICSUSER	YES	NO	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	Produ
G6UPB2ELPLOW7UHP	PLEX2	MV2D	PENFOLD	CICSDL2	IYK4ZDL2	CICSUSER	YES	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT		CICSPSB	CICSRES	CICSTRN	CICSTST	Produ
G6UPB2ELPLOW7UHP	PLEX2	MV2D	KEITHH	IYK3ZKHA	IYK3ZKHA	KEITHH	NO	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
G6UPB2ELPLOW7UHP	PLEX2	MV2D	PENFOLD	CICSDL1	IYK4ZDL1	CICSUSER	YES	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	Produ
G6UPB2ELPLOW7UHP	PLEX2	MV2D	GB1220	D2	IYK4ZON2	CICSUSER	YES	YES	YES	SHOW	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
MI1OZ9L671MFMGCT	PLEX2	MV2D	JATP	JATP1950	JATP1950	CICSUSER	NO	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
MI1OZ9L671MFMGCT	PLEX2	MV2D	GB1220	D1	IYK4ZON1	CICSUSER	YES	YES	YES	SHOW	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
MI1OZ9L671MFMGCT	PLEX2	MV2D	GB1220	D2	IYK4ZON2	CICSUSER	YES	YES	YES	SHOW	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
MI1OZ9L671MFMGCT	PLEX2	MV2D	JAMA	JATP0650	JATP0650	JAT237	NO	YES	YES	HIDE	JA09CMD		JA09DCT	JA09FCT	JA09JCT	JA09PCT	JA09PPT	JA09PSB	JA09RES	JA09TRN	JA09TST	Testir
MI1OZ9L671MFMGCT	PLEX2	MV2D	PENFOLD	CICSDL1	IYK4ZDL1	CICSUSER	YES	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	Produ
MI1OZ9L671MFMGCT	PLEX2	MV2D	KEITHH	IYK3ZKHA	IYK3ZKHA	KEITHH	NO	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
MI1OZ9L671MFMGCT	PLEX2	MV2D	KEITHH	JATP1550	JATP1550	CICSUSER	NO	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	
MI1OZ9L671MFMGCT	PLEX2	MV2D	ALHUNTE	CMASOVC1	CAHCOVC1	ALHUNTE	NO	YES	YES	HIDE						CICSPCT			CICSRES			
MI1OZ9L671MFMGCT	PLEX2	MV2D	PENFOLD	CICSDL2	IYK4ZDL2	CICSUSER	YES	YES	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT		CICSPSB	CICSRES	CICSTRN	CICSTST	Produ
MI1OZ9L671MFMGCT	PLEX2	MV2D	PENFOLD	CICSDL3	IYK4ZDL3	CICSUSER	YES	NO	YES	HIDE	CICSCMD		CICSDCT	CICSFCT	CICSJCT	CICSPCT	CICSPPT	CICSPSB	CICSRES	CICSTRN	CICSTST	Produ

- Identifies ALL regions on selected LPARs
- No configuration required
- CPSM not required

Using spreadsheet means easy to sort and identify non-conforming regions

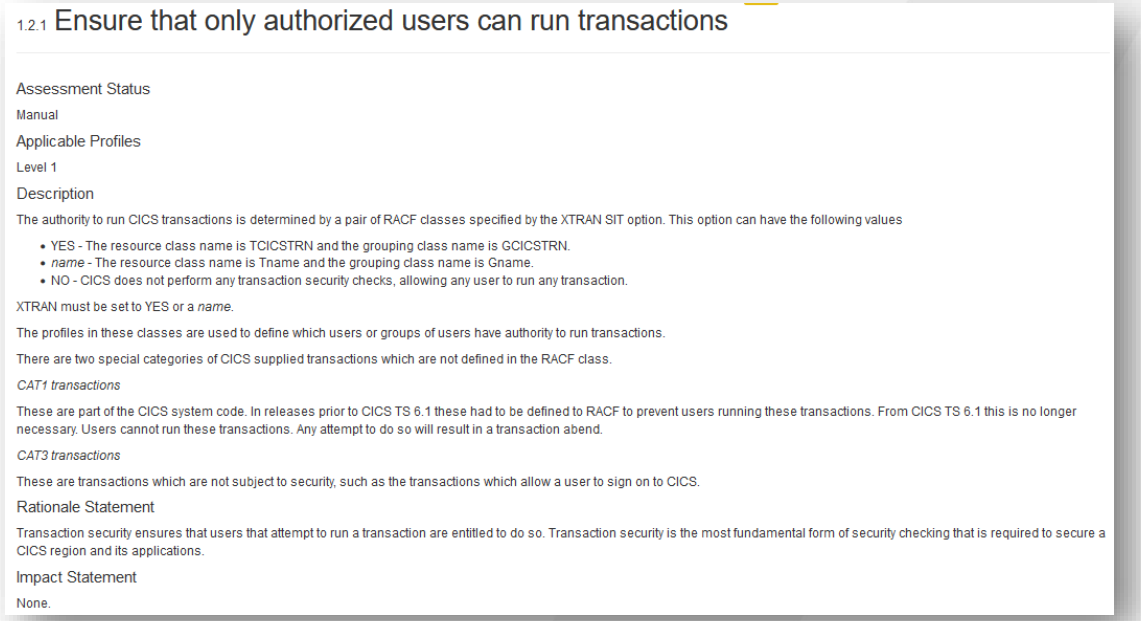
CICS Compliance Data – Tag & PTFs

	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ
	RegionUsage	RegionTypes	Applications	Other Tags	Release	PTF1		PTF2		PTF3		PTF4		PTF5
ST					60100	I2203193	20220322	I2403232	20220325					
					60100	I2503193	20220325	HDGFCAHI	20220328					
ST	Production	FOR	SCC	Owner:Colin_Penfold Stuff1:Stuff_1a Stuff1:Stuff_1b Stuff2:Stuff_2a Stuff2:Stuff_2b Stuff2:Stuff_2c	60100	I2503193	20220325							
ST	Production	AOR	SCC	Owner:Colin_Penfold Stuff1:Stuff_1a Stuff1:Stuff_1b Stuff2:Stuff_2a Stuff2:Stuff_2b Stuff2:Stuff_2c	60100	I2503193	20220325							
ST					60100	I1903222	20220319							
ST	Production	TOR	SCC	Owner:Colin_Penfold Stuff1:Stuff_1a Stuff1:Stuff_1b Stuff2:Stuff_2a Stuff2:Stuff_2b Stuff2:Stuff_2c	60100	I2503193	20220325							
ST					60100	I2203193	20220322							
ST					60100	I3003183	20220330							
ST					60100	I2203193	20220322	I2403232	20220325					
ST					60100	I2203193	20220322							
ST	Testing Creating	TOR	ExampleApp	L:UK	60100	I3003183	20220330							
ST	Production	TOR	SCC	Owner:Colin_Penfold Stuff1:Stuff_1a Stuff1:Stuff_1b Stuff2:Stuff_2a Stuff2:Stuff_2b Stuff2:Stuff_2c	60100	I2503193	20220325							
ST					60100	I1903222	20220319							
ST					60100	I3003183	20220330							
					60100	I2503193	20220325	HDGFCAHI	20220328					
ST	Production	AOR	SCC	Owner:Colin_Penfold Stuff1:Stuff_1a Stuff1:Stuff_1b Stuff2:Stuff_2a Stuff2:Stuff_2b Stuff2:Stuff_2c	60100	I2503193	20220325							
ST	Production	FOR	SCC	Owner:Colin_Penfold Stuff1:Stuff_1a Stuff1:Stuff_1b Stuff2:Stuff_2a Stuff2:Stuff_2b Stuff2:Stuff_2c	60100	I2503193	20220325							

- Region tags
- Allows auditor to identify production regions and their usage

- 5 most recent CICS runtime PTFs loaded
- Objective is to show running version is same as the SMP/E target library

CICS Compliance Data – Benchmarks



1.2.1 Ensure that only authorized users can run transactions

Assessment Status

Manual

Applicable Profiles

Level 1

Description

The authority to run CICS transactions is determined by a pair of RACF classes specified by the XTRAN SIT option. This option can have the following values

- YES - The resource class name is TCICSTRN and the grouping class name is GCICSTRN.
- *name* - The resource class name is Tname and the grouping class name is Gname.
- NO - CICS does not perform any transaction security checks, allowing any user to run any transaction.

XTRAN must be set to YES or a *name*.

The profiles in these classes are used to define which users or groups of users have authority to run transactions.

There are two special categories of CICS supplied transactions which are not defined in the RACF class.

CAT1 transactions

These are part of the CICS system code. In releases prior to CICS TS 6.1 these had to be defined to RACF to prevent users running these transactions. From CICS TS 6.1 this is no longer necessary. Users cannot run these transactions. Any attempt to do so will result in a transaction abend.

CAT3 transactions

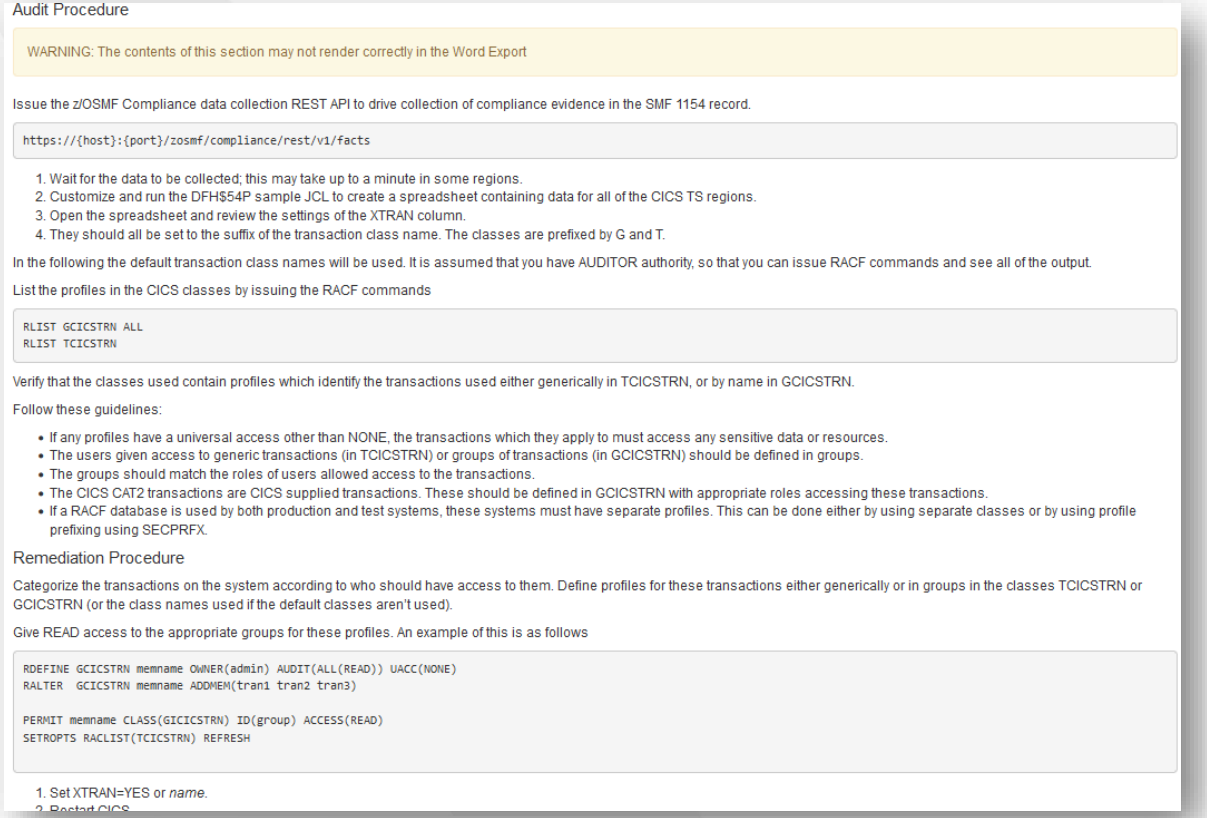
These are transactions which are not subject to security, such as the transactions which allow a user to sign on to CICS.

Rationale Statement

Transaction security ensures that users that attempt to run a transaction are entitled to do so. Transaction security is the most fundamental form of security checking that is required to secure a CICS region and its applications.

Impact Statement

None.



Audit Procedure

WARNING: The contents of this section may not render correctly in the Word Export

Issue the z/OSMF Compliance data collection REST API to drive collection of compliance evidence in the SMF 1154 record.

```
https://{host}:{port}/zosmf/compliance/rest/v1/facts
```

1. Wait for the data to be collected; this may take up to a minute in some regions.
2. Customize and run the DFH$54P sample JCL to create a spreadsheet containing data for all of the CICS TS regions.
3. Open the spreadsheet and review the settings of the XTRAN column.
4. They should all be set to the suffix of the transaction class name. The classes are prefixed by G and T.

In the following the default transaction class names will be used. It is assumed that you have AUDITOR authority, so that you can issue RACF commands and see all of the output.

List the profiles in the CICS classes by issuing the RACF commands

```
RLIST GCICSTRN ALL
RLIST TCICSTRN
```

Verify that the classes used contain profiles which identify the transactions used either generically in TCICSTRN, or by name in GCICSTRN.

Follow these guidelines:

- If any profiles have a universal access other than NONE, the transactions which they apply to must access any sensitive data or resources.
- The users given access to generic transactions (in TCICSTRN) or groups of transactions (in GCICSTRN) should be defined in groups.
- The groups should match the roles of users allowed access to the transactions.
- The CICS CAT2 transactions are CICS supplied transactions. These should be defined in GCICSTRN with appropriate roles accessing these transactions.
- If a RACF database is used by both production and test systems, these systems must have separate profiles. This can be done either by using separate classes or by using profile prefixing using SECPRFX.

Remediation Procedure

Categorize the transactions on the system according to who should have access to them. Define profiles for these transactions either generically or in groups in the classes TCICSTRN or GCICSTRN (or the class names used if the default classes aren't used).

Give READ access to the appropriate groups for these profiles. An example of this is as follows

```
RDEFINE GCICSTRN memname OWNER(admin) AUDIT(ALL(READ)) UACC(NONE)
RALTER  GCICSTRN memname ADDMEM(tran1 tran2 tran3)

PERMIT memname CLASS(GICICSTRN) ID(group) ACCESS(READ)
SETROPTS RACLIST(TCICSTRN) REFRESH
```

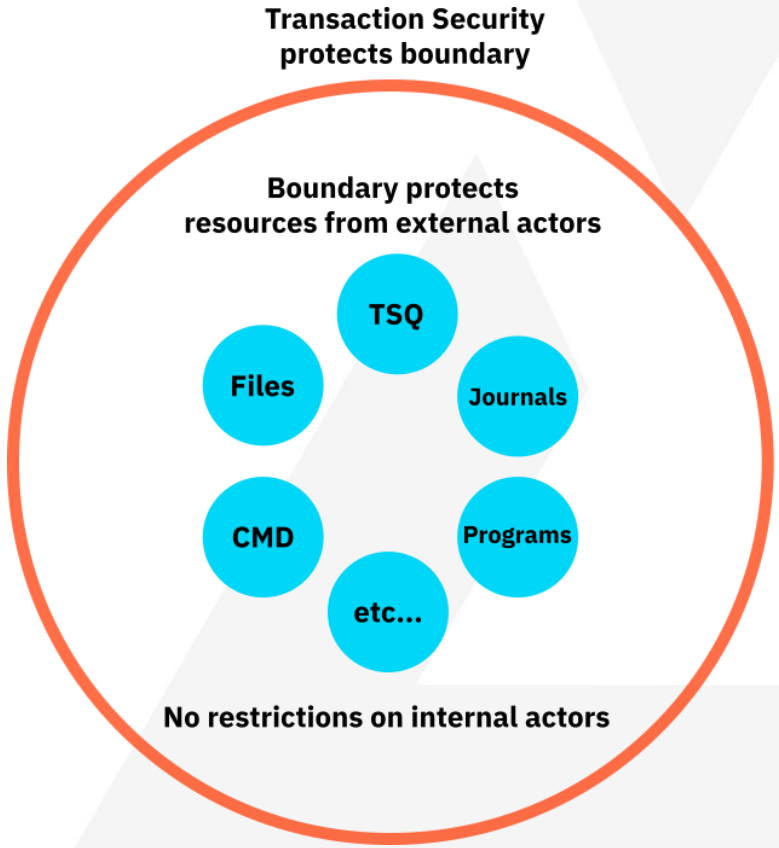
1. Set XTRAN=YES or *name*.
2. Restart CICS

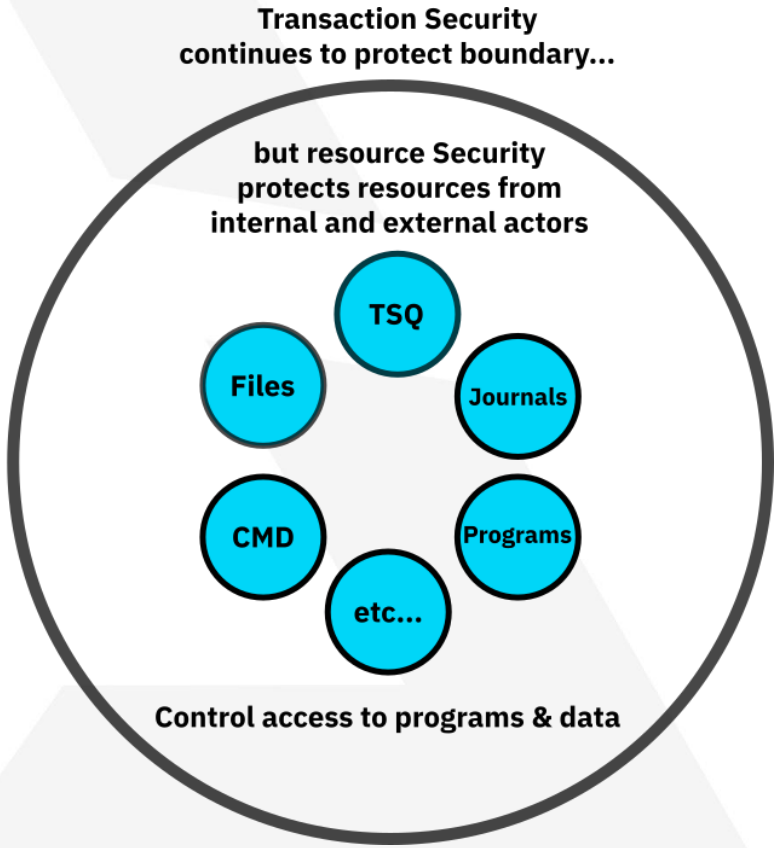
https://workbench.cisecurity.org/benchmarks/9876

CICS Security Discovery

Traditional transaction security is no longer considered ‘secure enough’.

- Industry wide push to enabling resource security – but it's not that easy





CICS Security Discovery

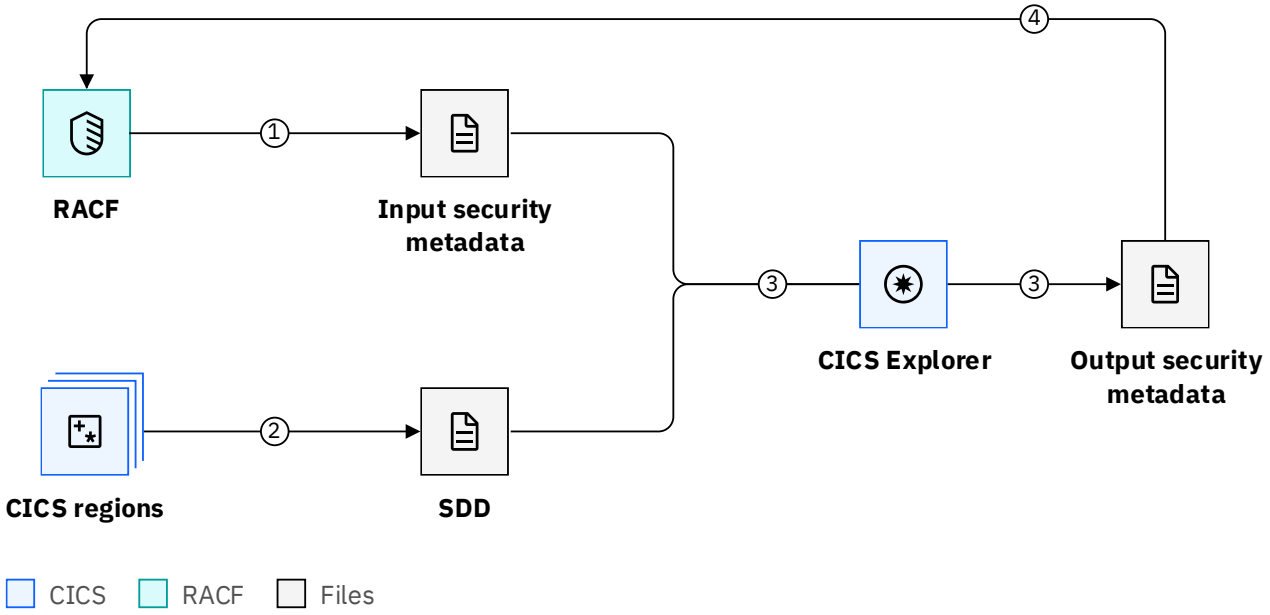
The difficulty in enabling resource security comes down to key challenges

- Most customers do not have sufficient security testing
- Customers do not have effective role-based security access
- It's difficult to know which user needs access to which new definitions

Migrating to resource security without breaking mission critical applications

CICS Security Discovery

A tool to log ‘would be’ security access in production before resource security is turned on



1. Extract RACF definitions

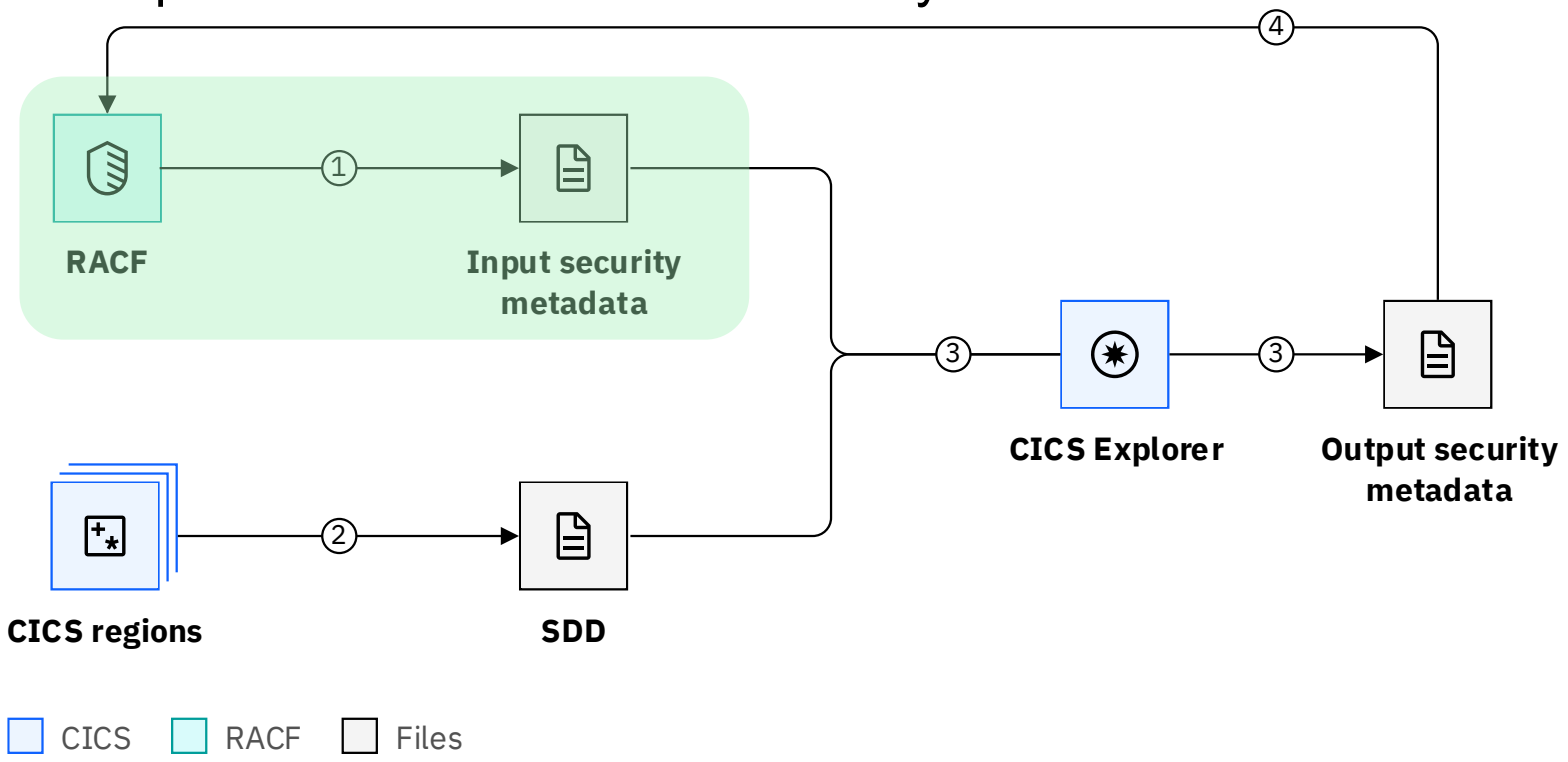
2. Capture security discovery data (SDD)

3. Analyze security definitions in CICS Explorer and export them for review

4. Create RACF commands from reviewed security definitions

CICS Security Discovery

Import RACF definitions as Security Metadata



DFH$R2SM

```
//SECMETA JOB
//SECMETA  EXEC DFHXSMET,SAMPLIB=hlq.SDFHSAMP,
//      DIR='/u/userid/cics/', FN='regionsA'
//INPUT.SYSIN DD *
XTRAN=CICSTRN
SECPRFX=NO
//
```

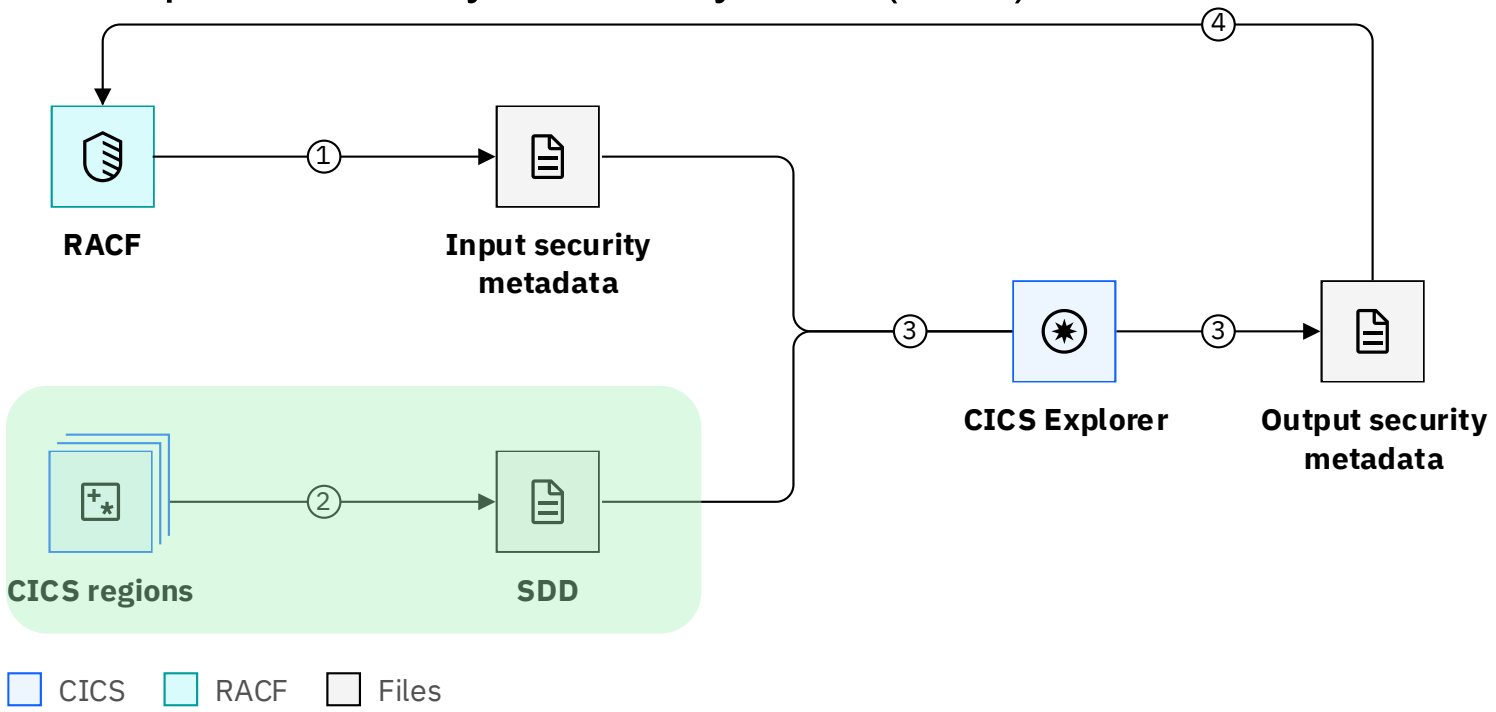
Security Metadata

```
--- # Security Metadata ---
version: 2
file_created:
  - date: "17 Mar 2023"
  - time: "17:27:26"
  - user: SUE
group_list:
  - name: MANAGER
    users:
      - MAINWRN
  - name: TELLER
    users:
      - WILSON
      - PIKE
user_list:
  - user: JONES
    username: "Jack Jones"
  - user: MAINWRN
    username: "George Mainwaring"
  - user: PIKE
    username: "Frank Pike"
  - user: WILSON
    username: "Arthur Wilson"
secprfx: NO
classes:
  - class: XTRAN
    name: CICSTRN
    profiles:
      - name: BANKING
        members:
          - BNK1
          - BNK2
        access_lists:
          - access: READ
            groups:
              - MANAGER
              - TELLER
            users:
              - JONES
```

- → Imports transaction class and groups
- → Assumes users have good transaction security
- → Separate import for each SECPRFX
- → Optionally imports other CICS security classes

CICS Security Discovery

Capture Security Discovery Data (SDD)



SPI commands

```
SET SECDISCOVERY

 < STATUS() | ON | OFF >
 < DISCOVERALL | _classes_ >

INQUIRE SECDISCOVERY

 < STATUS() >
 < TRAN() > _classes_
 < LASTSECDTIME() > < LASTWRITTIME() >
 < SECDCOUNT() > < NEWSECDCOUNT() >

PERFORM SECDISCOVERY WRITE
```

DFH$SDDP

```
//SECDSDD JOB
...
//SDD      EXEC PGM=DFHSDD
//STEPLIB  DD DISP=SHR,DSN=&HLQ..SDFHLINK
//DFHSECD  DD other definitions,DSNAME=&LOGSTRM
//DFHSDD   DD other definitions,PATH='&SDD'
...
```

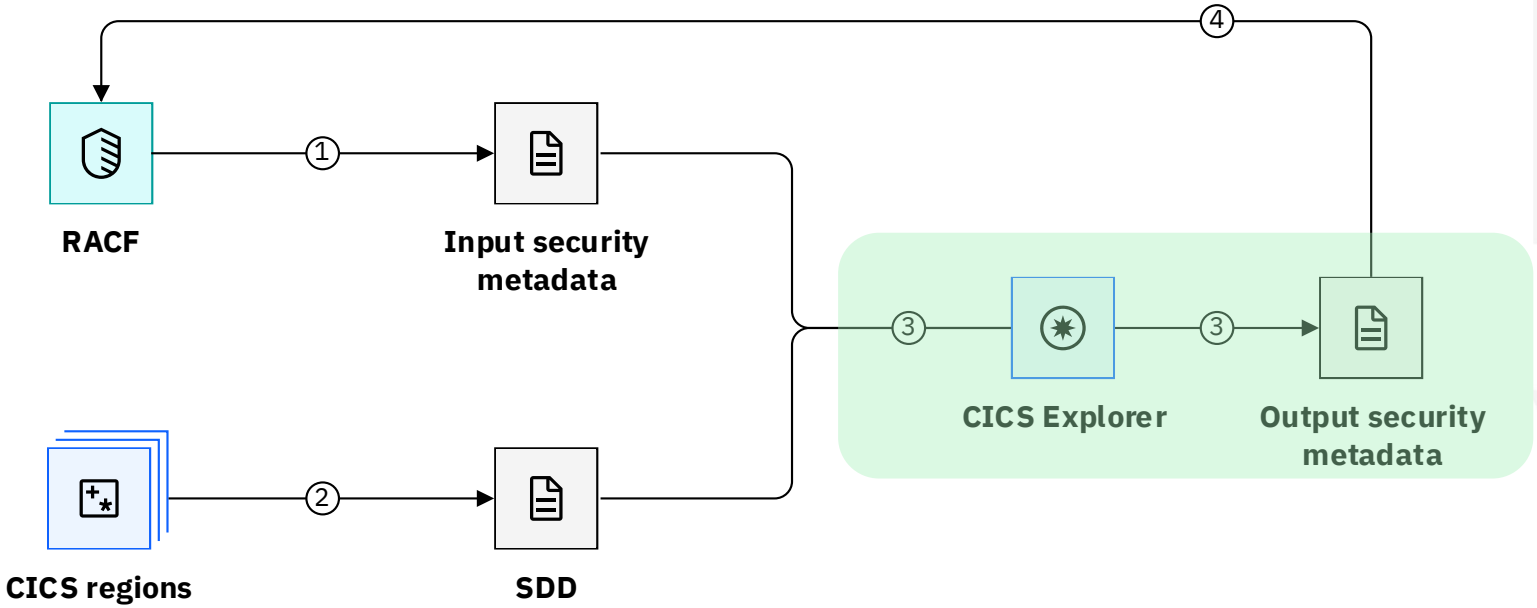
→ Switch on SECDISCOVERY to capture usage data written to logstream per day
→ Message DFHXS1602 shows new data captured each day
→ When there is little ‘new data’ being captured, process SDD

SDD

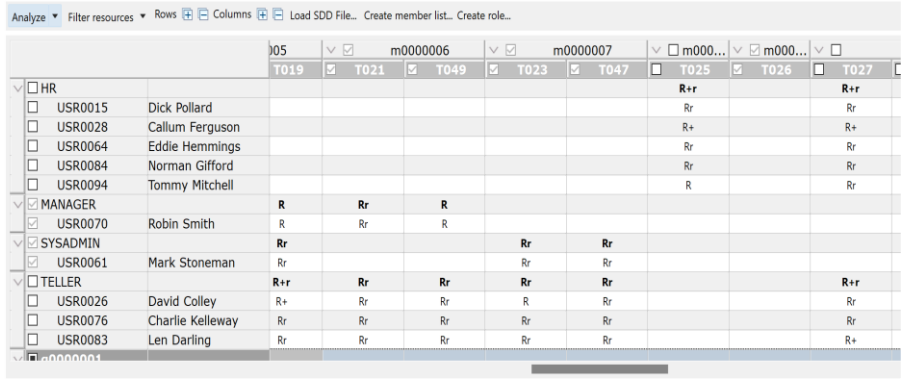
```
userid,principal,session,region,default,resource type,class,profile,max
access,READ,UPDATE,CONTROL,ALTER,INQUIRE,SET,COLLECT,PERFORM,EXECUTE,CREATE,DISCARD,DEFINE,INSTALL,START,SUBMIT,DE
LETE,SET_CONTROL,SET_ALTER,QUERYU0000001,Y,N,N,N,TRANSATTACH,TCICSTRN,T002,READ,N,N,N,N,N,N,N,Y,N,N,N,N,N,N,N,N,
N,NU0000001,Y,N,N,N,TRANSATTACH,TCICSTRN,T009,READ,N,N,N,N,N,N,N,Y,N,N,N,N,N,N,N,N,NU0000002,Y,N,N,N,TRANSATTA
CH,TCICSTRN,T001,READ,N,N,N,N,N,N,N,Y,N,N,N,N,N,N,N,N,NU0000002,Y,N,N,N,TRANSATTACH,TCICSTRN,T002,READ,N,N,N,N
,N,N,N,N,Y,N,N,N,N,N,N,N,N,NU0000003,Y,N,N,N,TRANSATTACH,TCICSTRN,T004,READ,N,N,N,N,N,N,N,Y,N,N,N,N,N,N,N,N,
NU0000003,Y,N,N,N,TRANSATTACH,TCICSTRN,T005,READ,N,N,N,N,N,N,N,Y,N,N,N,N,N,N,N,N,N
```

CICS Security Discovery

Analyze security definitions in CICS Explorer and export them for review



CICS Explorer– Security Discovery Perspective



Use Security Metadata to:

→ Identify user groups (roles) and their access to transaction member lists

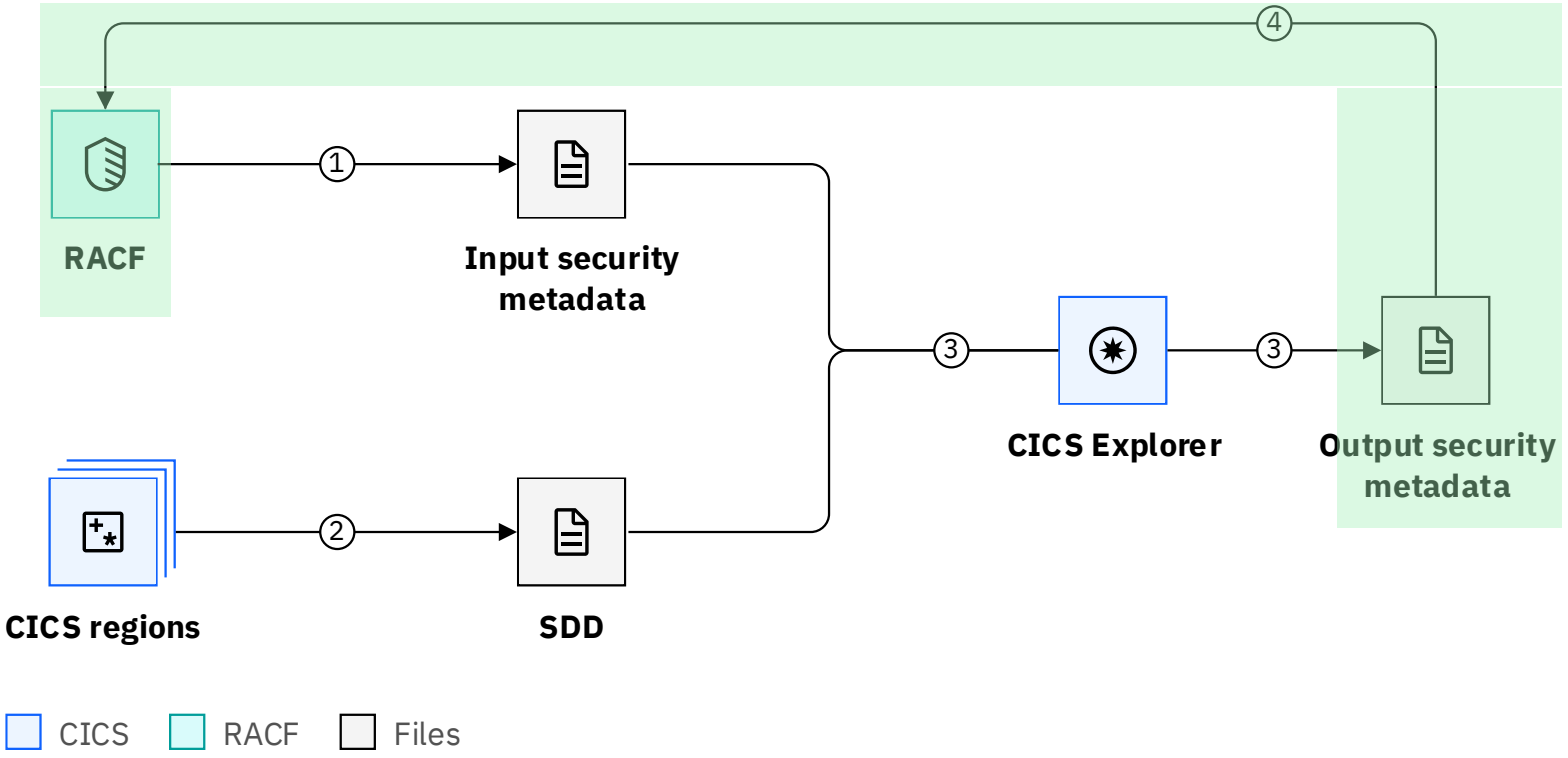
Use Security Discovery Data to:

→ Refine user groups and transaction member lists

→ Identify user groups access to resource member lists

CICS Security Discovery

Create RACF commands from review



DFH$SM2R

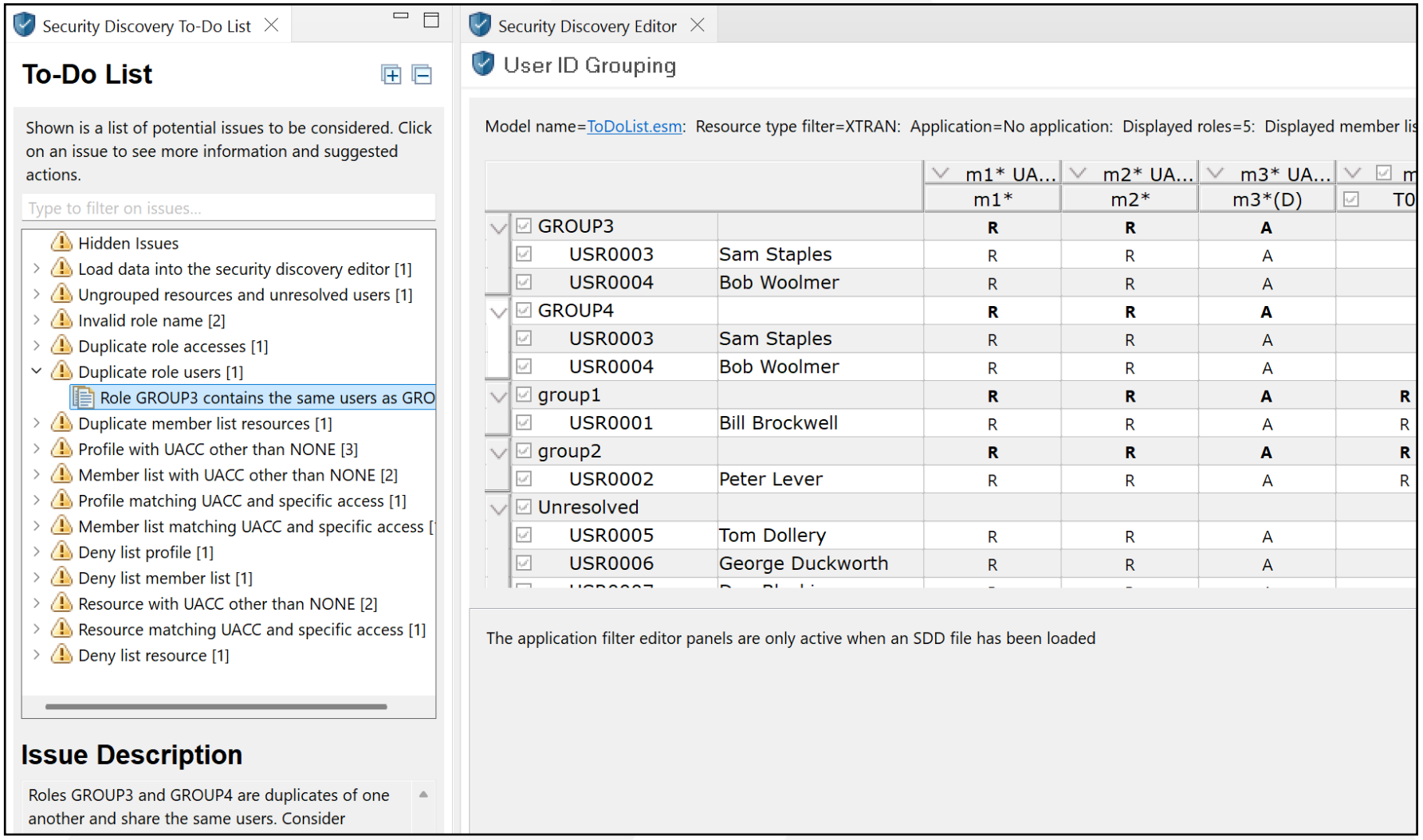
```
//SECMETA JOB REGION=0M
//RACFCMD  EXEC PGM=IRXJCL,PARM='DFH$XSR'
//SYSEXEC  DD DISP=SHR,DSN=hlq.SDFHSAMP
//YAML     DD DISP=SHR,DSN=<<security
metadata>>
//XTRAN    DD SYSOUT=*
//XFCT     DD SYSOUT=*
...
//GROUPS   DD SYSOUT=*
//SYSIN    DD *
XTRAN=TESTTRN
XFCT=TESTFCT
//
```

RACF commands

```
RDEFINE GCIC1TRN TGRP1 +
                       UACC(NONE) +
                       ADDMEM(TRNA,TRNB,TRNC,TRND)
PERMIT TGRP1 CLASS(GCIC1TRN)  +
                       ACCESS(READ) +
                       ID(UGRP1)
RDEFINE GCIC1TRN TGRP2 +
                       UACC(NONE) +
                       ADDMEM(TRNE,TRNF)
PERMIT TGRP1 CLASS(GCIC1TRN)  +
                       ACCESS(READ) +
                       ID(UGRP2)
```

- → Creates new classes (allows easy migration)
- → Configure for selected regions at a time
- → Options to customize with WARNING, OWNER, etc

CICS Security Discovery



SHARE
EDUCATE • NETWORK • INFLUENCE

For a deep dive on CICS Security Discovery

CICS Security Discovery

Thursday 26th February 2026 09:15am

Security Definition Capture (SDC)

Customer research - 2023

Resource security is not enabled for most customers

- Cost and complexity
- Unknowing where to start

Developers will often go through System Programmers to get security access

- Slow process

Pre-production tends to be the earliest part of security testing

- Limited automation

Security Definition Capture (SDC)

https://devclass.com/2023/01/26/devsecops-report/

"*the biggest barrier to DevSecOps being that security teams do not trust developers, identified by 55 percent of organizations as the top issue*" *

* Based on a survey of 1300 DevOps and Security professionals in large enterprises

Security Definition Capture (SDC)

How can we enable customers to take advantage of the benefits of DevSecOps without having extensive DevOps pipelines

[+] A manual solution is required to bridge the gap between customers today and a full automation solution in the future

[+] An automation solution to be documented with a sample solution for those customers with good practice pipelines

Security Definition Capture (SDC)

Identify security definitions during ‘business as usual’ application testing whether:

- Manual, 3270 terminal interaction
- Fully fledged automation

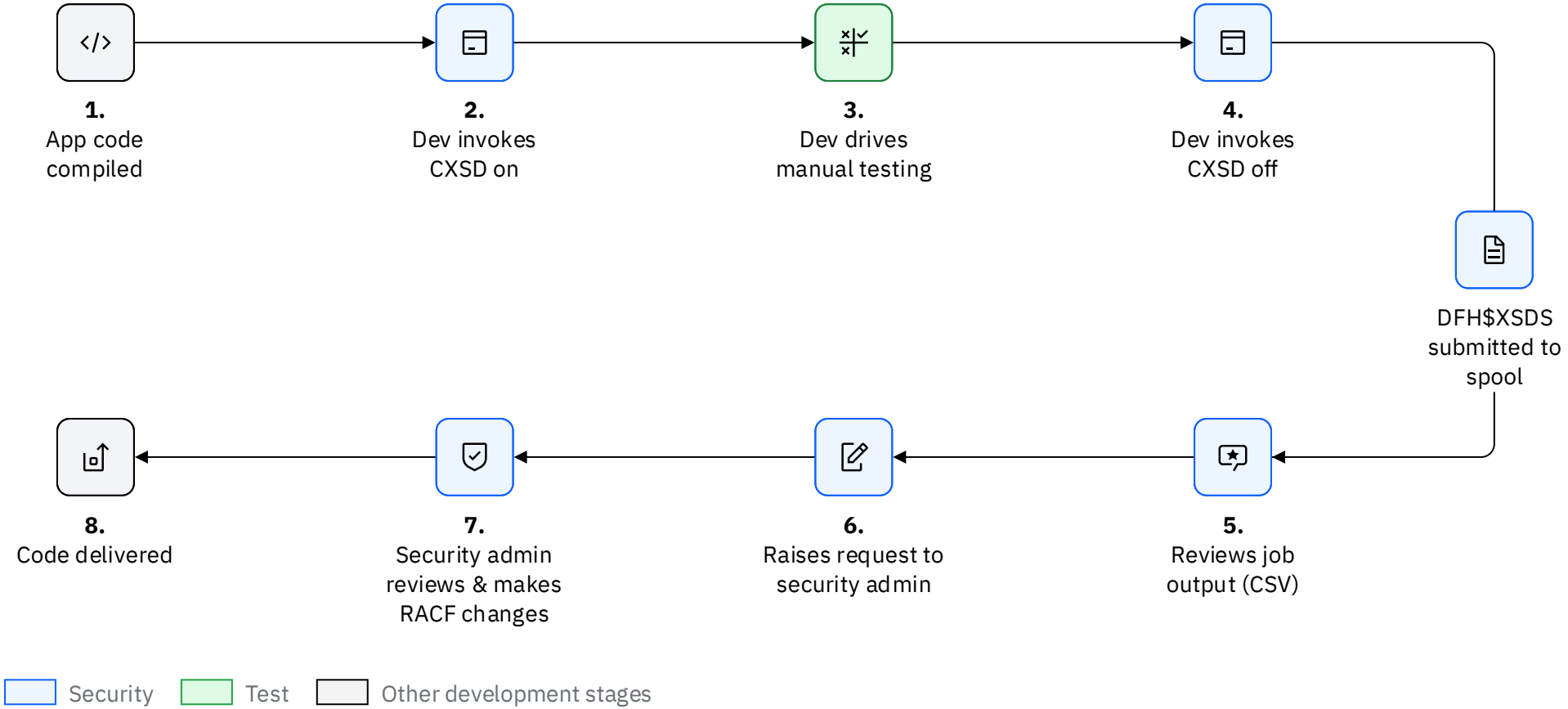
Using development systems using minimal security (SEC=YES)

No security expertise needed of the user

Security requests captured in the background with no extra effort

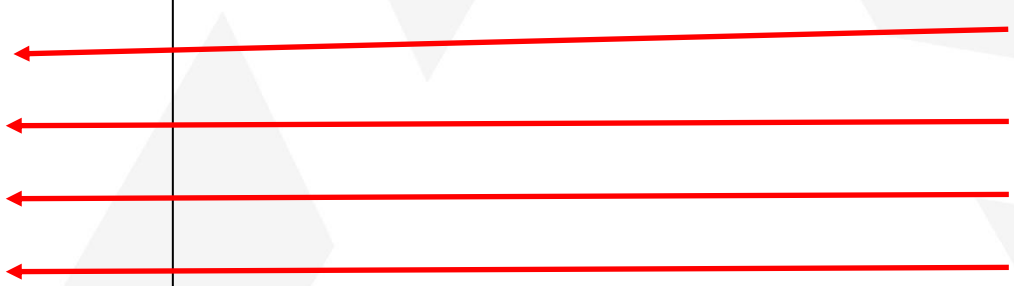
1. HTTP Restful Interface to toggle SDC on/off
2. Terminal transaction CXSD toggled before and after testing

Security Definition Capture (SDC)



Security Definition Capture (SDC)

```
--- # Security Metadata
version: 1
classes:
 - name: XCMD
   profiles:
    - name: FILE
      access_lists:
       - access: READ
         users:
          - U000125
 - name: XFCT
   profiles:
    - name: FILEA
      access_lists:
       - access: READ
         users:
          - U000125
    - name: FILEB
      access_lists:
       - access: READ
         users:
          - U000125
 - name: XTRAN
   profiles:
    - name: TRNA
      access_lists:
       - access: READ
         users:
          - U000125
```



The resource type

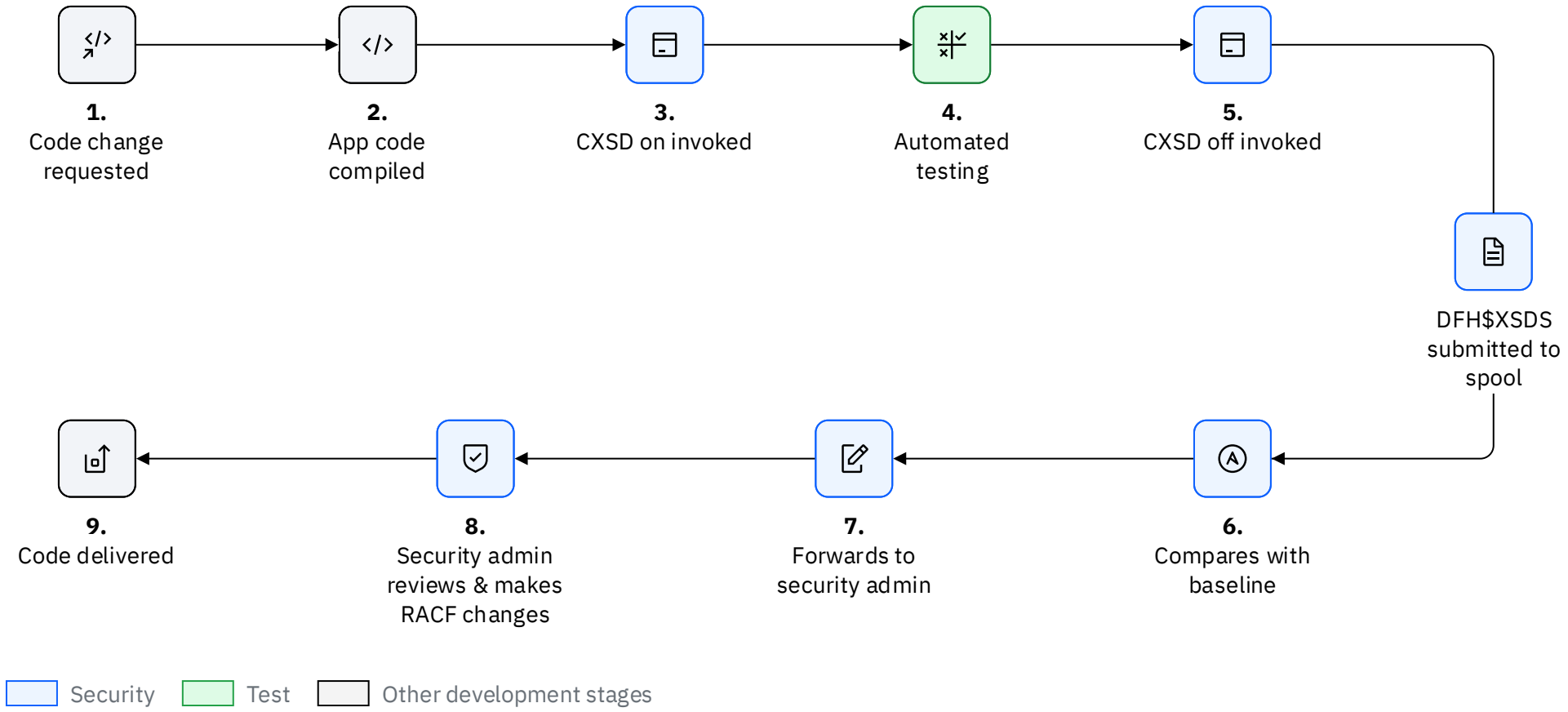
File name

Access level

User name

The user name is that of the developer/tester. They would represent a user role for the production security definition.

Security Definition Capture (SDC) - DevOps



Security Definition Capture (SDC)

Before code change

```
--- # Security Metadata
version: 1
classes:
 - name: XCMD
   profiles:
    - name: FILE
      access_lists:
       - access: READ
         users:
          - U000125
 - name: XFCT
   profiles:
    - name: FILEA
      access_lists:
       - access: READ
         users:
          - U000125
    - name: FILEB
      access_lists:
       - access: READ
         users:
          - U000125
 - name: XTRAN
   profiles:
    - name: TRNA
      access_lists:
       - access: READ
         users:
          - U000125
```

After code change

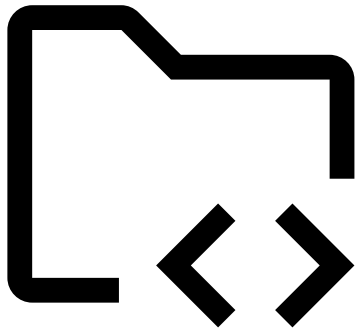
```
--- # Security Metadata
version: 1
classes:
 - name: XCMD
   profiles:
    - name: FILE
      access_lists:
       - access: READ
         users:
          - U000125
 - name: XFCT
   profiles:
    - name: FILEA
      access_lists:
       - access: READ
         users:
          - U000125
    - name: FILEB
      access_lists:
       - access: UPDATE
         users:
          - U000125
 - name: XTRAN
   profiles:
    - name: TRNA
      access_lists:
       - access: READ
         users:
          - U000125
```

Security changes required

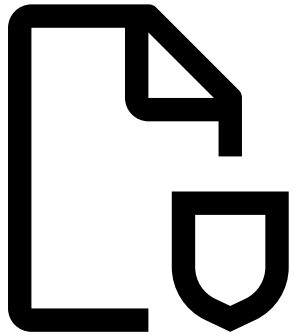
```
 - name: FILEB
        access_lists:
D -       - access: READ
I -       - access: UPDATE
          users:
           - U000125
```

Security Definition Capture (SDC)

Source Control Repository



`application.cbl`



`security.esm`

Security Definition Capture (SDC)

Implementing SDC into a development pipeline.

Open Mainframe Project – Galasa Test Framework

- Implementation of an open-source framework manager

Enables the easy implementation of SDC into existing test suites with minimal effort

Developer sample and blog from CICS team:

https://cicsdev.github.io/cics-security-sdv-samples/

Changes to RESSEC and CMDSEC

Zero Trust and Compliance with PCI-DSS

It is recommended that customers use resource and command security for production regions to secure sensitive data.

Before CICS TS 6.2 – RESSEC(NO CMDSEC(NO) default

RESSEC=ALWAYS, CMDSEC=ALWAYS SIT parameters should be set

However.

Many CICS transactions are defined with RESSEC(NO) CMDSEC(NO).

There are implications for CICS and Customer transactions.

Security definitions may need to be defined.

Changes to RESSEC and CMDSEC

Changing CICS Definitions

All CICS transactions in 6.2 have been changed to RESSEC(YES) CMDSEC(YES)

→ Many do not have resource security checks

→ Some unnecessary security checks removed (trusted applications)

Existing definitions added to compatibility group DFHCOMPK

Why do some CICS transactions not require security checking

CICS code is trusted as long as its totally encapsulated

Signoff transaction (CESF) is trusted

Commands issued by CECI are not trusted

→ CESF writes a message to CSMT TDQ

→ Pointless requiring security definitions for CESF to do this.

Changes to RESSEC and CMDSEC

Changing Customer Definitions

RESSEC(YES) and CMDSEC(YES) are the new defaults for transaction definitions

Existing transaction definitions are unchanged

Use Security Discovery to implement RACF definitions before changing either RESSEC or CMDSEC

After all transactions are changed consider either:

Setting SIT parameters RESSEC=ALWAYS and CMDSEC=ALWAYS

Using CICS Resource Overrides to ensure new transactions being defined conform to this policy

Health Checker

- CICS introduced an initial 3 sets of health checks in CICS TS 5.4
- Designed to advise best practice based on production-like environments
- No configuration required
- CICS 6.1 introduces 5 new sets of health checks:
  - **CICS_CAT3_CONFIGURATION**
  - **CICS_REGION_CONFIGURATION**
  - **CICS_RESOURCE_CONFIGURATION**
  - **CICS_RESOURCE_SECURITY**
  - **CICS_USS_CONFIGURATION**

Health Checker

```
SDSF HEALTH CHECKER DISPLAY  MV2C                    LINE 3-25 (255)
COMMAND INPUT ===>                                      SCROLL ===> DATA
PREFIX=*  DEST=(ALL)  OWNER=DBEARD*  SYSNAME=
NP    NAME                          CheckOwner       State              Status              Result Diag1    Diag2    DiagFrom Gl
      ALLOC_SPEC_WAIT_POLICY        IBMALLOC         ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      ALLOC_TAPELIB_PREF            IBMALLOC         ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      ALLOC_TIOT_SIZE               IBMALLOC         ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      ASM_LOCAL_SLOT_USAGE          IBMASM           ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      ASM_NUMBER_LOCAL_DATASETS     IBMASM           ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      ASM_PAGE_ADD                  IBMASM           ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      ASM_PLPA_COMMON_SIZE          IBMASM           ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      ASM_PLPA_COMMON_USAGE         IBMASM           ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      CATALOG_ATTRIBUTE_CHECK       IBMCATALOG       ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      CATALOG_IMBED_REPLICATE       IBMCATALOG       ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CATALOG_RNLS                  IBMCATALOG       ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      CICS_CAT3_CONFIGURATION       IBMCICS          ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CICS_CEDA_ACCESS              IBMCICS          ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CICS_JOBSUB_SPOOL             IBMCICS          ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CICS_JOBSUB_TDQINTRDR         IBMCICS          ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
      CICS_REGION_CONFIGURATION     IBMCICS          ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CICS_RESOURCE_CONFIGURATION   IBMCICS          ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CICS_RESOURCE_SECURITY        IBMCICS          ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CICS_USS_CONFIGURATION        IBMCICS          ACTIVE(ENABLED)    EXCEPTION-LOW            4 00000000 00000000          NO
      CNZ_AMRF_EVENTUAL_ACTION_MSGS IBMCNZ           ACTIVE(ENABLED)    SUCCESSFUL               0 00000000 00000000          NO
```

Health Checks added

Pre-6.1 CICS Health Checks

Health Checker

```
 SDSF OUTPUT DISPLAY CICS_USS_CONFIGURATION        LINE 44      COLUMNS 02- 133
 COMMAND INPUT ===>                                            SCROLL ===> PAGE

01/25/2022 12:53:52.313896 CIDRBAF1 0061 IYK2ZAF1 DBEARD1  0740 C000 13
Exception messages:
DFHH0601 Unauthorised access to USSCONFIG is allowed.
DFHH0602 Unauthorised access to JVMPROFILE is allowed.

01/25/2022 12:56:56.863431 IYK3ZDD6 0068 IYK3ZDD6 DONNELL  0740 8000 675
Exception messages:
DFHH0601 Unauthorised access to USSCONFIG is allowed.

01/25/2022 12:58:52.468779 CIDRBAF1 0061 IYK2ZAF1 DBEARD1  0740 C000 14
Exception messages:
DFHH0601 Unauthorised access to USSCONFIG is allowed.
DFHH0602 Unauthorised access to JVMPROFILE is allowed.
```

Health Checker

IBM provides the Health Checker for z/OS as ‘free advice’

Middleware products ‘plug in’ to the checker for domain specific areas

- Recent internal IBM data collection found 30% override and turn it off…
- Health checker promotes best practice aligning with zero trust
- Certain checks can be suppressed where appropriate using Region Tagging

Query Security

```
EXEC CICS QUERY SECURITY LOGMESSAGE
```

- Query the security authorization of a user to access a resource.

`NOLOG` option means no security violation messages issued.

- This could be exploited by bad user.
- Scanning what access, they can exercise

Query Security

CICS TS 6.2 provides new statistics and monitoring fields

Enable system admin can be aware of how many times security violation happened without message issued.

Statistics:

- XSG_AUTHOR_FAIL_NL_NA - Failed authorizations NOLOG NOTAUTH
- XSG_AUTHOR_FAIL_NL_NF - Failed authorizations NOLOG NOTFND

Monitoring fields:

- XSNLNACT
- XSNLNFCT

Conclusion

Adhering to all the security regulations is hard.

- Regulations are consistently getting stronger

A Zero Trust strategy encompasses the key behaviors required to be compliant

- Shares common ground across most global regulations

IBM has built tools in CICS TS 6 to enable the journey to be easier

There is no overnight answer, customers need to be proactive not reactive

What we need from you

Here at the conference
In sessions, conversations, and later via email when you've had time to digest presentation material

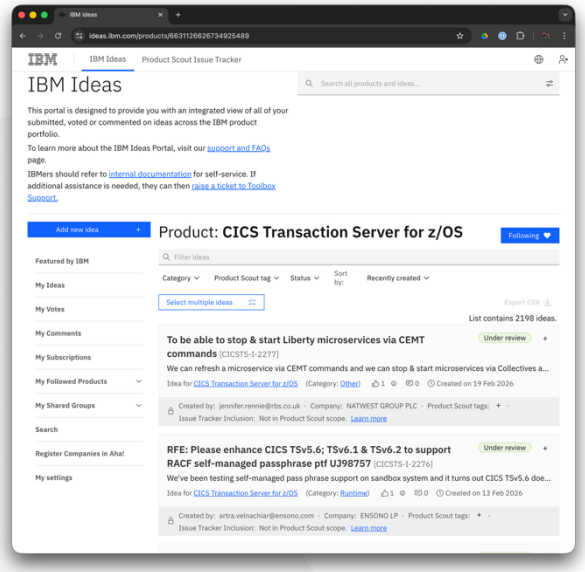
Join the CICS Design Partnership or business partner program

Learn about upcoming features from our design and development teams.

Give your feedback during presentations, polls, 1-1 interviews

CICS Community in TechXchange: blogs, threads, materials, events
IBM CICS TS group on LinkedIn: recent announcement, events, and though-provoking posts
CICS tag on StackExchange: Q&A about CICS
CICS-L list forum: a non-IBM, moderated community of experienced CICS users asking each other questions

Raise an Idea!
Search, add your vote, subscribe to updates, discuss alternatives, add your use cases in the comments, raise new ideas, feedback from CICS team – all on the IBM Ideas Portal



Your feedback is important!

SHARE
EDUCATE • NETWORK • INFLUENCE

Submit a session evaluation for each session you attend:

www.share.org/evaluation