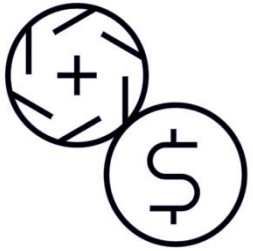
SHARE Winter 2026 – Orlando, FL

SHARE
EDUCATE • NETWORK • INFLUENCE

CICS TS 6 - Performance Update

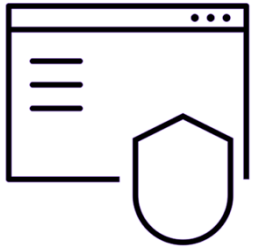
Lewis James
CICS TS for z/OS Development
IBM UK

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

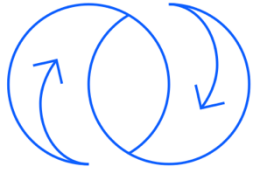


Improved security and compliance management

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

CICS TS 6 Introduces Rich Functionality

Developer Experience & Modernization

- Support for Java 17 & 21
- CMCI JVMServer
- Ansible automation

System Management

- IBM Health Checker for z/OS increased CICS checks
- CICS Region Tagging
- Support for OpenTelemetry
- CICS MCP server

Trust and Compliance

- Support for TLS 1.3
- Security Request Recording
- Security Discovery & Security Definition Capture (Zero Trust)

To only mention just a few…

All working towards the theme of simplification and modernization

The Key Pillars of z/OS

It's all about the reputation

Unmatched reliability and availability

- The Famous Nines (99.999%)

Enterprise grade security

- Integration with highly securable ESM's (RACF, TopSecret, ACF2)

High volume transactional throughput

- High value, low-latency transactional workloads.
- Transactions per second

Adopting new functionality

IBM encourages customers to take strategic advantage in adopting new function

- Involves making investment in upgrading CICS
  - Not a simple task

CICS TS 6 aims to simplify, modernize, and make things generally easier

Providing tools for diagnosis, compliance, and system management

Generally making things better but at what cost?

How Performance is Measured

Measurements between the CICS 6 releases automated on controlled environments

Workload run on an IBM z16

- Maximum of 32 CP's (central processor) available to the LPAR
- Internal CF co-located in same CP complex as measurement LPARs
- IBM System Storage DS8950F provides external storage
- LPARs on z/OS 2.5

Comparisons run on GA versions of the release code

RELEASE TO RELEASE COMPARISON

Data System Workload (DSW)

Performance Testing uses Data System Workload (DSW)

- A non-threadsafe COBOL application which uses VSAM files
- Transactions use BMS to interface with 3270 terminals
  - Number of terminals used was 4,000
- Number of files used was 32
- DSW uses a number of transactions where 50% issue at least 1 file control request. On average the are 6 file control requests per task with a breakdown as follows:-
  - 69% read
  - 10% read for update
  - 9% update
  - 11% add
  - 1% delete

ETR – External Throughput Rate : the total number of transaction completions per second in the TORs

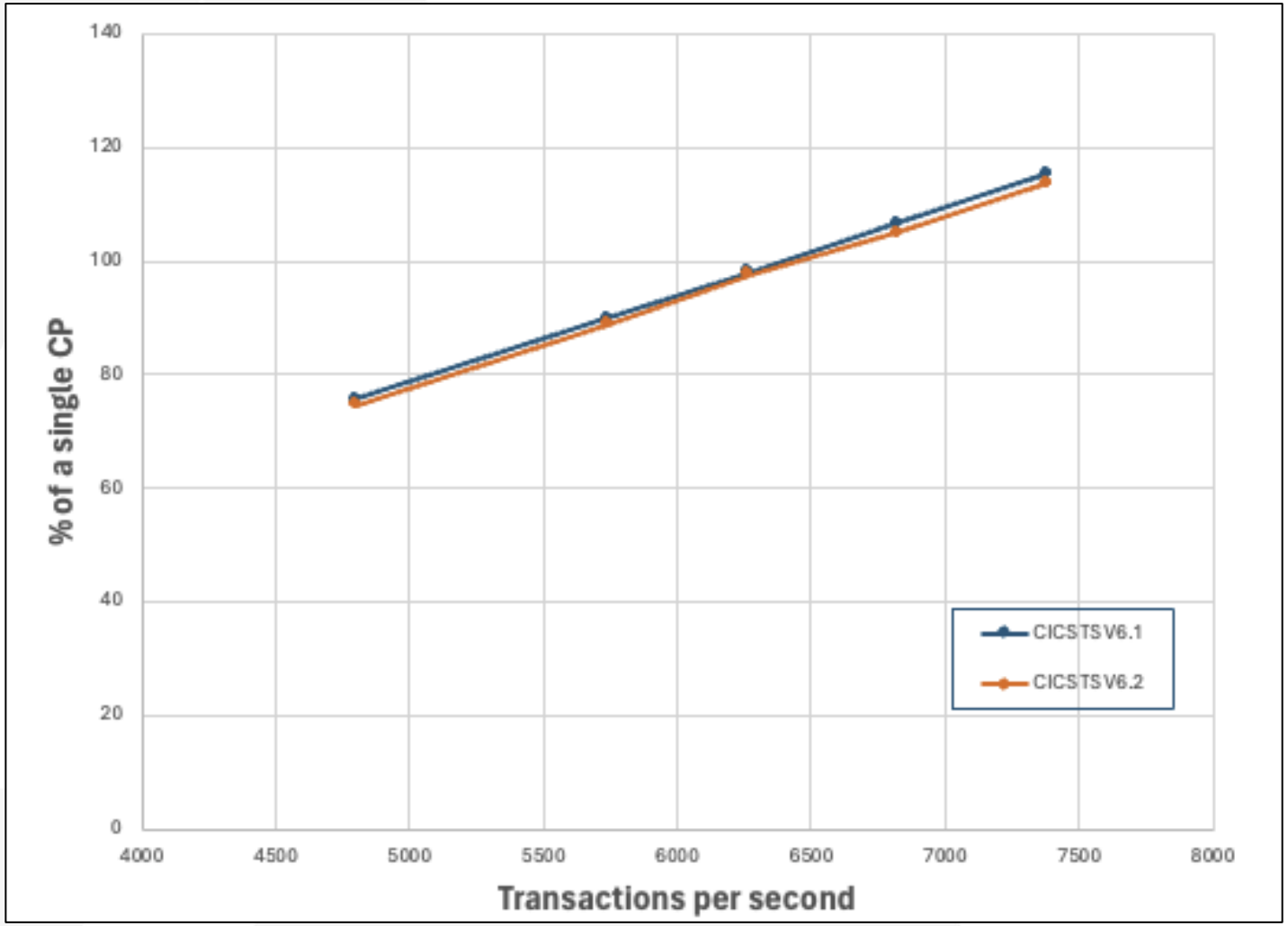
DSW Static Routing Workload 6.1 – 6.2

ETR	CICS CPU (%)	CPU per Transaction (ms)
4801.38	75.70	0.158
5739.77	89.97	0.157
6269.04	98.10	0.156
6821.59	106.63	0.156
7385.76	115.37	0.156

ETR	CICS CPU (%)	CPU per Transaction (ms)
4800.91	74.63	0.155
5739.52	88.77	0.155
6268.57	96.70	0.154
6821.69	105.07	0.154
7384.05	113.73	0.154

DSW Static Routing Workload 6.1 – 6.2

- The calculated CPU per transaction rate for the releases differed by less than 1%



Relational Transactional Workload (RTW)

- A COBOL application that accesses an IBM Db2 database
- 7 transaction types access 16 Db2 tables by using EXEC SQL commands
- In total, slightly more than 30 million rows of data are defined in the database
- On average each CICS task issues 200 Db2 SQL calls with the following percentages
  - 54% SELECT
  - 1% INSERT
  - 1% UPDATE
  - 1% DELETE
  - 8% OPEN CURSOR
  - 27% FETCH CURSOR
  - 8% CLOSE CURSOR

RTW Workload 6.1 – 6.2

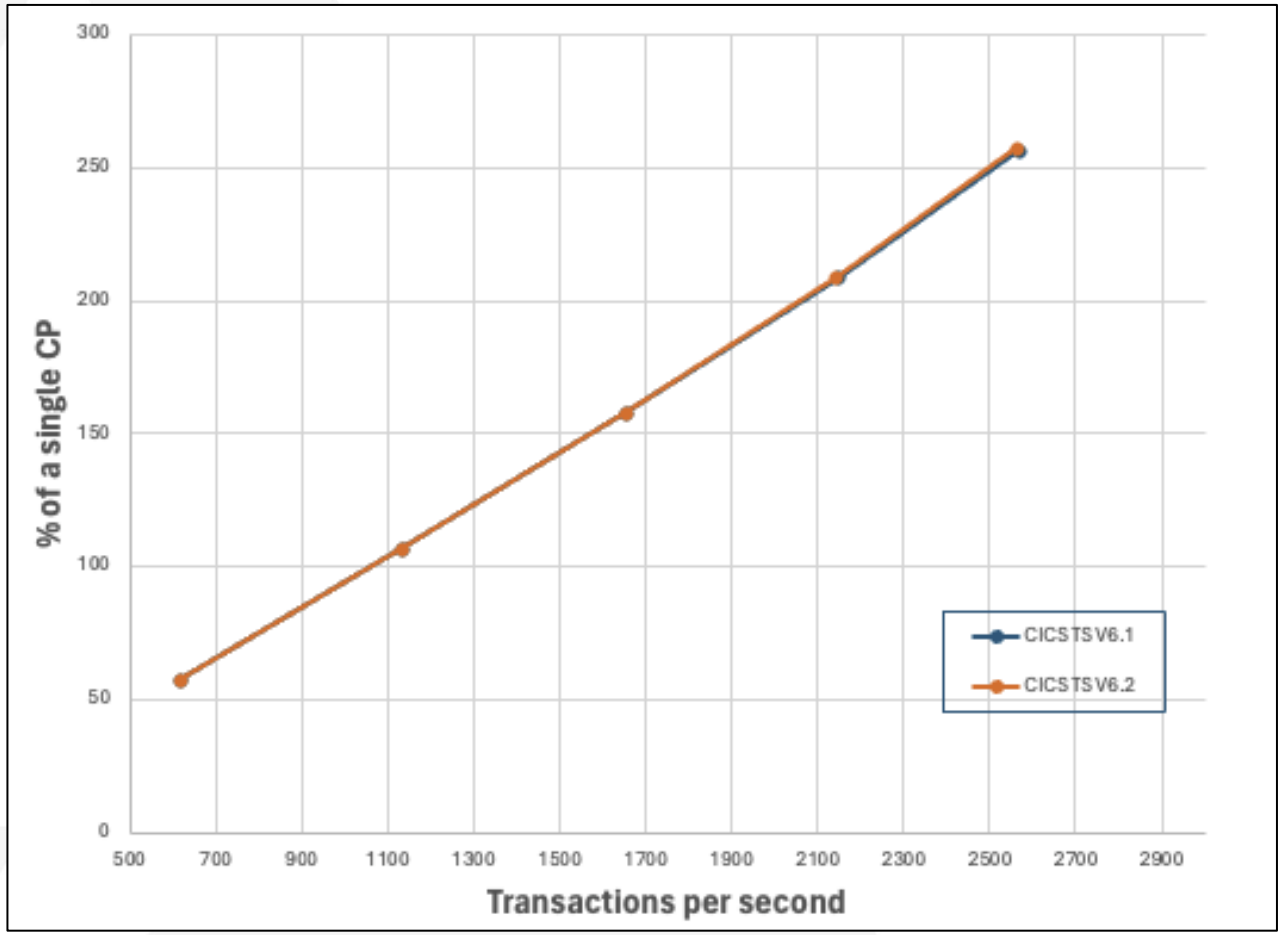
ETR	CICS CPU (%)	CPU per Transaction (ms)
616.89	57.13	0.926
1132.80	106.43	0.940
1655.10	158.10	0.955
2145.36	208.57	0.972
2567.66	256.50	0.999

ETR	CICS CPU (%)	CPU per Transaction (ms)
616.88	57.10	0.926
1132.73	106.50	0.940
1655.04	158.10	0.955
2144.17	209.03	0.975
2564.03	257.10	1.003

RTW Workload 6.1 – 6.2

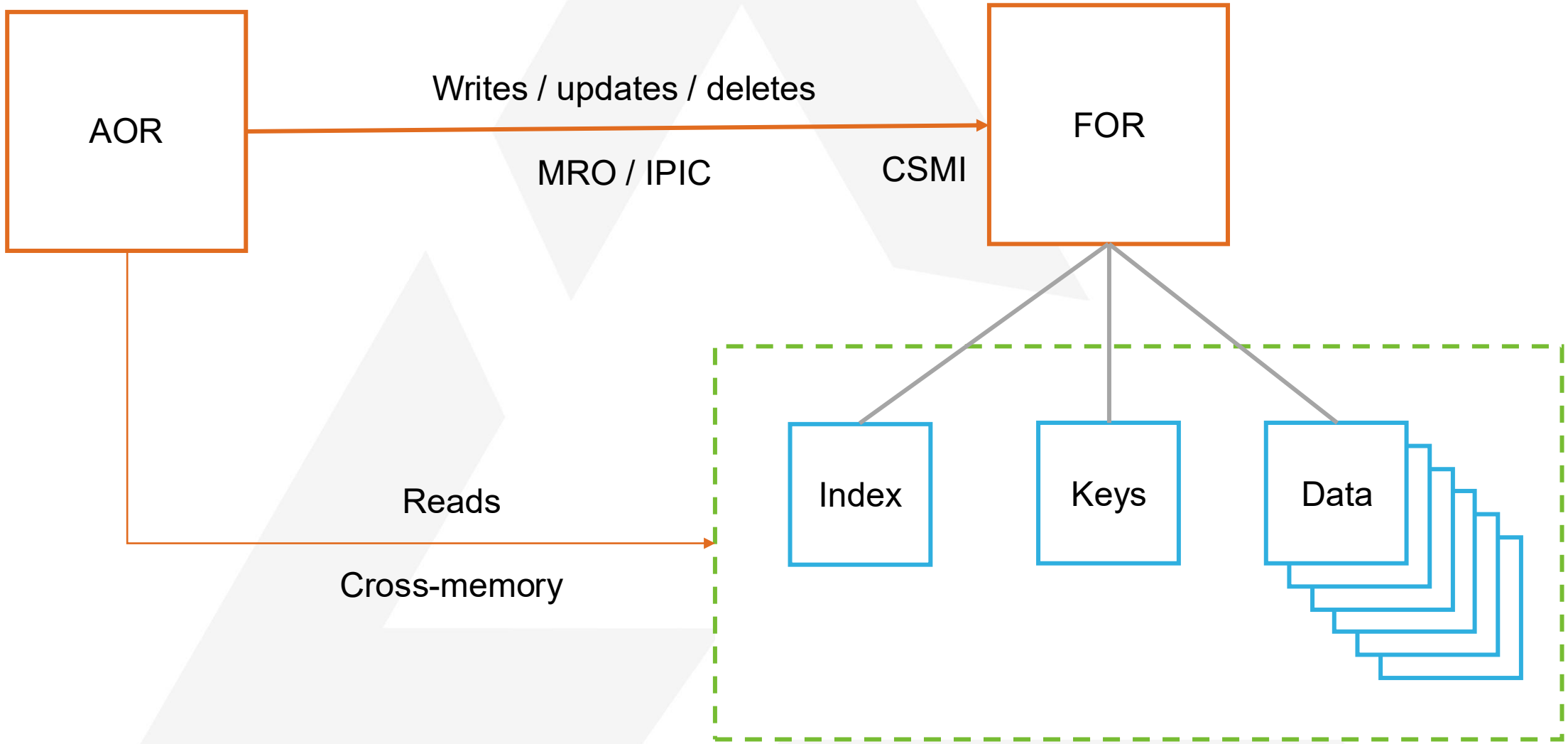
- CICS TS 6.1 contained all PTFs issued before 2024 Jan 29
- CICS TS 6.2 used GA level code
- Both releases were on z/OS 2.5

- Measurements show that the performance CICS TS 6.2 is equivalent to that of CICS TS 6.1 for this workload



SHARED DATA TABLES

Shared Data Table (SDT)



Shared Data Table Enhancements

Prior to CICS TS 6 we had the following architecture

- Single 2GB data space for index
- Single 2GB data space for keys

Limitation when some VSAM record lengths is made up of mostly the key

CICS TS for z/OS 6.1

Enhancement to use 64-bit storage to store index and key data

- Obtained from GCDSA
- SIT parameter SDTMEMLIMIT to prevent consumption of all storage
  - Defaults to 4GB

Shared Data Table Enhancements

Summary

- Moved SDT control data from 31-bit to 64-bit
- Added SDTMEMLIMIT capping above-the-bar usage

Effects of performance

No measurable increase in CPU per transaction

Enhancements raise the capacity not effecting cost

Shared Data Table Enhancements

Data Tables - Storage

Filename	Type	Current Records	Peak Records	<- Entries + Index --> Storage Allocated	Storage In-Use	<------ Entries -----> Storage Allocated	Storage In-Use	<------- Index ------> Storage Allocated	Storage In-Use	<------- Data -------> Storage Allocated	Storage In-Use
VSAM1S1A	USER	10,000,000	10,000,000	399,488	399,306	234,496	234,375	164,992	164,931	655,872	644,532
VSAM1S1B	USER	10,000,000	10,000,000	399,488	399,306	234,496	234,375	164,992	164,931	655,872	644,532
VSAM1S2A	USER	10,000,000	10,000,000	399,488	399,306	234,496	234,375	164,992	164,931	655,872	644,532
VSAM1S2B	USER	10,000,000	10,000,000	399,488	399,306	234,496	234,375	164,992	164,931	655,872	644,532
VSAM1S5A	USER	10,000,000	10,000,000	399,488	399,306	234,496	234,375	164,992	164,931	655,872	644,532
Totals				1,997,440	1,996,530	1,172,480	1,171,875	824,960	824,655	3,279,360	3,222,660

FILES - Data Table Requests Information

File Name	Close Type	Read Requests	Recs ^ in Table	Adds from Add Reads	Add Requests	<- Adds Rejected -> By Exit	Table Full	Rewrite Requests	Delete Requests	Highest Table Size	Storage Alloc(K)	Chng Resp/ Lock Waits
VSAM1S1A		188837	0	0	94919	0	0	93918	94919	10000044	1386112	0
VSAM1S1B		189607	0	0	94919	0	0	94688	94919	10000047	1388160	0
VSAM1S2A		192852	0	0	96096	0	0	96756	96096	10000042	1385344	0
VSAM1S2B		190388	0	0	96096	0	0	94292	96096	10000043	1387264	0
VSAM1S5A		188547	0	0	95183	0	0	93366	95182	10000053	1385600	0
TOTALS		950231	0	0	477213	0	0	473020	477212	10000053		0

Shared Data Table Enhancements

Multiple options for interacting with SDT:

READ, UPDATE, REWRITE, WRITE, DELETE

Prior to 6.2 AOR SDT READs non-threadsafe

- Forces task onto the QR TCB (even if application threadsafe)
- Creating contention and bottleneck during high volume

During high volume periods caused throughput capping despite plenty of CPU capacity.

Shared Data Table Enhancements

CICS TS 6.2 enables read-only SDT interaction to be threadsafe

- Data integrity is not a concern for read-only function

- The AOR uses file control to directly resolve the record from in-memory table
  - No function shipping
  - No switching to the QR TCB – remains on open TCB

[+] Reduces QR TCB contention for heavy-read volume

[+] Free up the FOR for modification operations

SDT Threadsafe Performance

Performance Topology

- [2x **TOR**, 5x **AOR**, 2x **FOR**]
- Each FOR hosted 5 SDT files [**klen**: 4 bytes] [**rlen**: 80 bytes]
- Each file initialized with 10,000,000 records loaded into main storage

Workload Driven

- 2,000 simulated terminals triggering transactions on TOR's
- Each transaction touches ten records across two SDT files

95% **READ**, 2% **READ UPDATE**, 1% **REWRITE**, 1% **DELETE**

SDT Threadsafe Performance

Performance Observations

Non threadsafe performance testing

ETR	CPU per transaction in AOR (ms)	CPU per transaction in FOR (ms)	Response time (ms)
996	0.070	0.003	0.22
2000	0.069	0.003	0.32
3070	0.068	0.003	0.41
3995	0.068	0.003	0.49
5000	0.068	0.003	0.64

CICS TS 6.1

ETR	CPU per transaction in AOR (ms)	CPU per transaction in FOR (ms)	Response time (ms)
996	0.069	0.003	0.21
2000	0.067	0.002	0.30
3060	0.066	0.003	0.38
3995	0.066	0.003	0.51
5000	0.066	0.003	0.61

CICS TS 6.2

SDT Threadsafe Performance

Performance Observations

Threadsafe performance testing

ETR	CPU per transaction in AOR (ms)	CPU per transaction in FOR (ms)	Response time (ms)
996	0.080	0.003	0.26
2000	0.078	0.003	0.39
3070	0.077	0.003	0.51
3995	0.077	0.003	0.68
5000	0.077	0.003	0.86

CICS TS 6.1

ETR	CPU per transaction in AOR (ms)	CPU per transaction in FOR (ms)	Response time (ms)
996	0.076	0.003	0.24
2000	0.073	0.003	0.36
3070	0.073	0.003	0.46
3994	0.071	0.003	0.64
5000	0.071	0.003	0.80

CICS TS 6.2

Conclusion

For non-threadsafe applications:

- An effectively equal measurement between 6.1 and 6.2

For threadsafe applications:

- At 6.1 prior to threadsafe read…
  - Peak throughput of 17,500 transactions per second
  - Response time ~ 16 ms
  - QR dispatch at 100%
- At 6.2 with threadsafe read…
  - Peak throughput of 19,800 transactions per second (+ 2,300 tps)
  - Response time ~ 2.4ms (- 13.6 ms)
  - QR dispatch at 76.75% (- 23.25%)

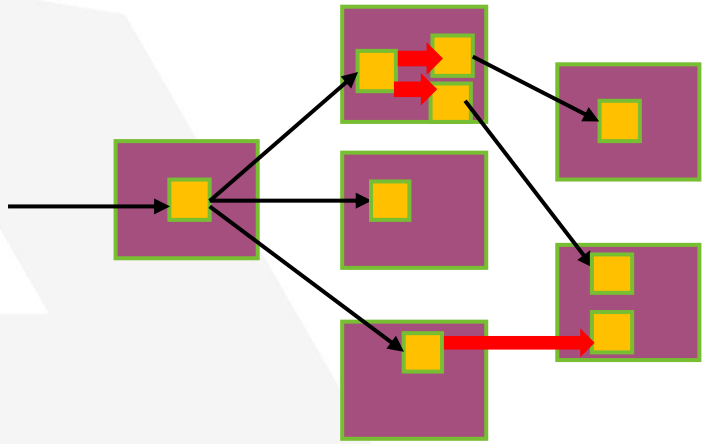
SECURITY REQUEST RECORDING

Security Request Recording

Introduced in CICS TS 6.1

Intended as a production diagnostic tool when security violations occur

- Aid in solving problems where complex security configuration through an application disguises where things are going wrong
- Reveal clues when `ICH408I` and `DFHXS1111` do not help



Security Request Recording

How it works

1. User hits a security violation error when running a transaction
2. User raises issue with security administrator
3. Security administrator creates a security request recording (SRR)
4. Recording is based on task origin data such as the origin user ID or txn

The user then reattempts the transaction and at each SAF call the origin data of the task is compared with the security request recording entries in the system

If it matches, the data of the SAF check is recorded in a logstream against the recording ID

Security Request Recording

Performance concerns

SRR is designed find the route cause of a security issue as quickly as possible.

- Required to be ran in production as often problem cannot be recreated in development

Customers often concerned what the performance overhead in running security request recording is

Even if CICS only using XTRAN – prospective resource checks also captured

Security Request Recording

Performance Topology

- [2x **TOR**, 2x **AOR**, 1x **FOR**]
- All running on single LPAR
- Work driven by DSW – excessive VSAM file interaction

Test 1

- Monitoring of workload performance across all regions with SRR disabled

Test 2

- SRR enabled with origin data on one ODUSER (FRED)
- User FRED not invoking the workload (on but not capturing)

Security Request Recording (cont.)

Test 3

- SRR enabled for all users (capturing every SAF call)

Test Case	TOR CP %	TOR tran/min	AOR CP %	AOR tran/min	FOR CP %	FOR tran/min
Test 1 (No SRR)	6.65%	115,500	16.65%	115,500	17.08%	231,000
Test 2 (User SRR)	7.10%	113,800	17.10%	113,800	17.05%	227,600
Test 3 (All User SRR)	7.39%	114,100	17.28%	114,100	17.40%	228,200

Security Request Recording

Performance conclusion

Targeted SRR (recommended usage) has negligible cost

- Running for specific user approx. 3-4µs per transaction

Capturing all users (larger recording criteria) raises cost

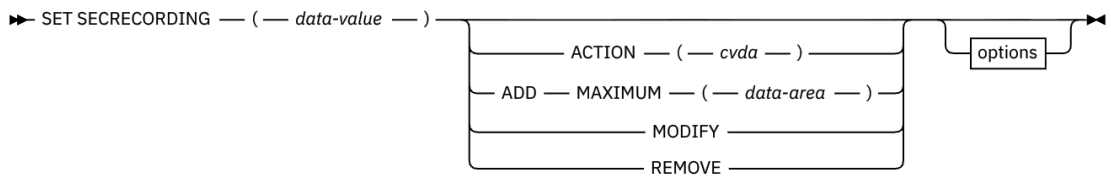
- 4.4µs per transaction in TOR & AOR
- 1.4µs per transaction in FOR
- Measurable but a small fraction of overall cost

Deployment settings matter

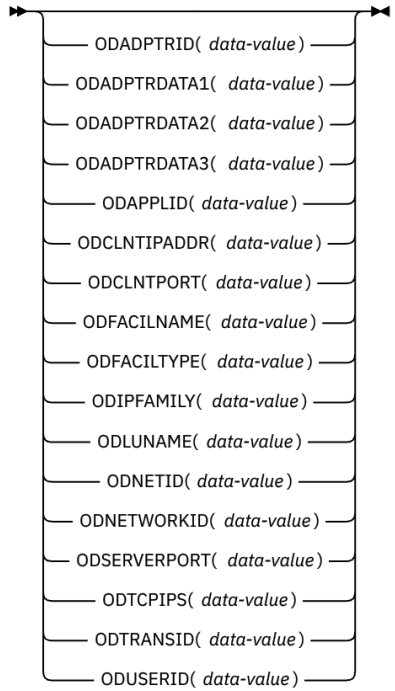
- SRR should be enabled in a targeted way, not capturing everything
- Only enable in the origin region such as TOR

Security Request Recording

SET SECRECORDING



Options



SECURITY DISCOVERY

Security Discovery

Tooling in CICS 6.2 enabling the discovery of security access driven in a production region.

Designed to enable the migration from transaction security to resource security

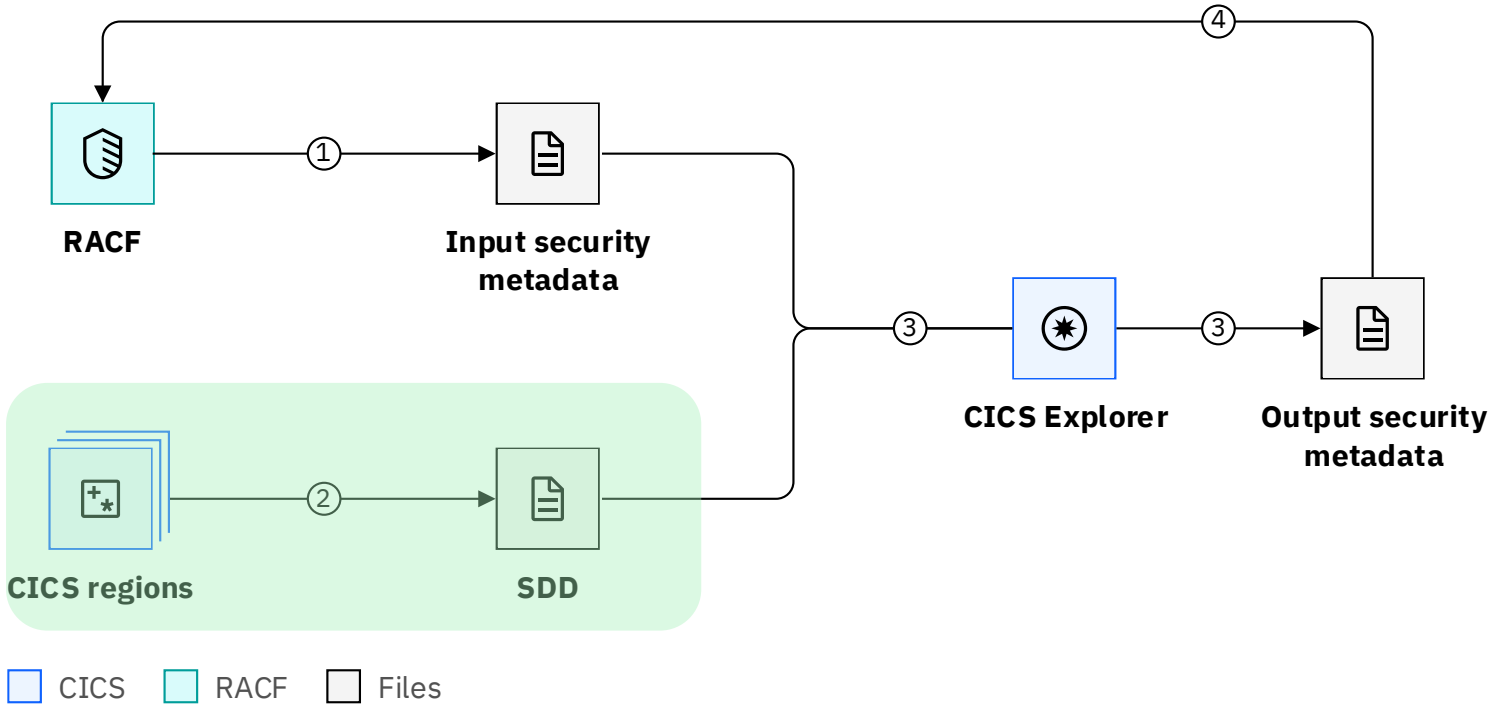
- Runs within a production CICS environment for a long period of time
- Captures when a user either has or would have been checked through SAF call

Invoked with the SPI provided

```
EXEC CICS SET SECDISCOVERY
```

Security Discovery

How it works



→ 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

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'
...
```

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,T009,READ,N,N,N,N,N,N,N,N,Y,N,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,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,N,Y,N,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,N,NU0000003,Y,N,N,N,TRANSATTACH,TCICSTRN,T004,READ,N,N,N,N,N,N,N,N,Y,N,N,N,N,N,N,N,N,N,
NU0000003,Y,N,N,N,TRANSATTACH,TCICSTRN,T005,READ,N,N,N,N,N,N,N,N,Y,N,N,N,N,N,N,N,N,N,N
```

Security Discovery

Performance concerns

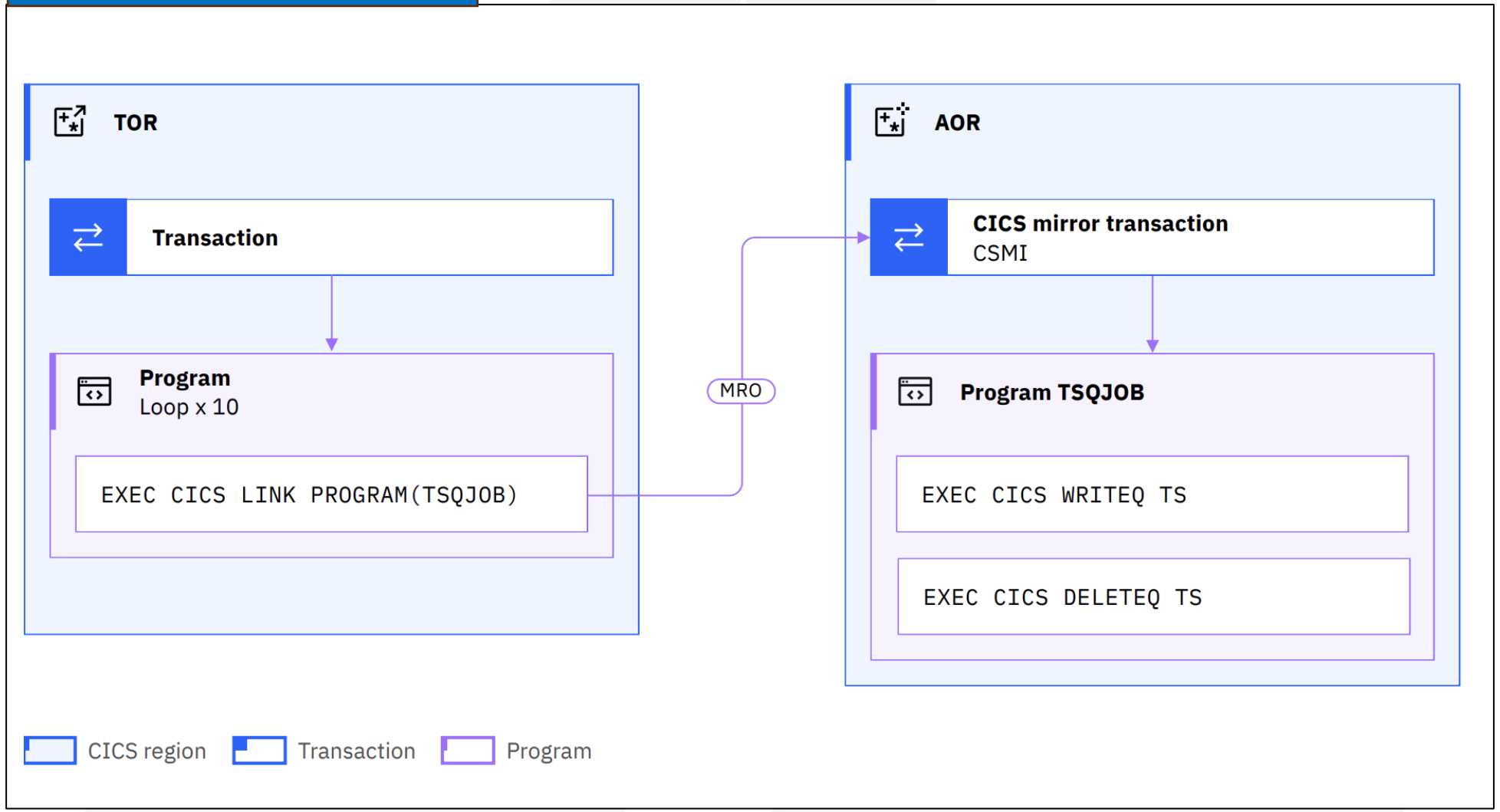
Security Discovery will capture information about every SAF call

Information must be gathered from a production region

Customers common questions relate to cost of discovering access requests

Security Discovery

Scenario 1



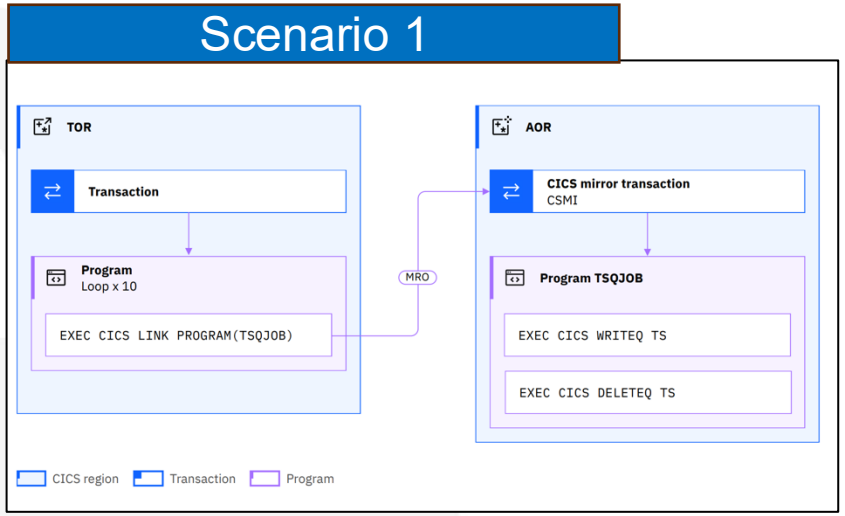
Security Discovery

Performance Testing

Test 1: Running workload with Security Discovery disabled

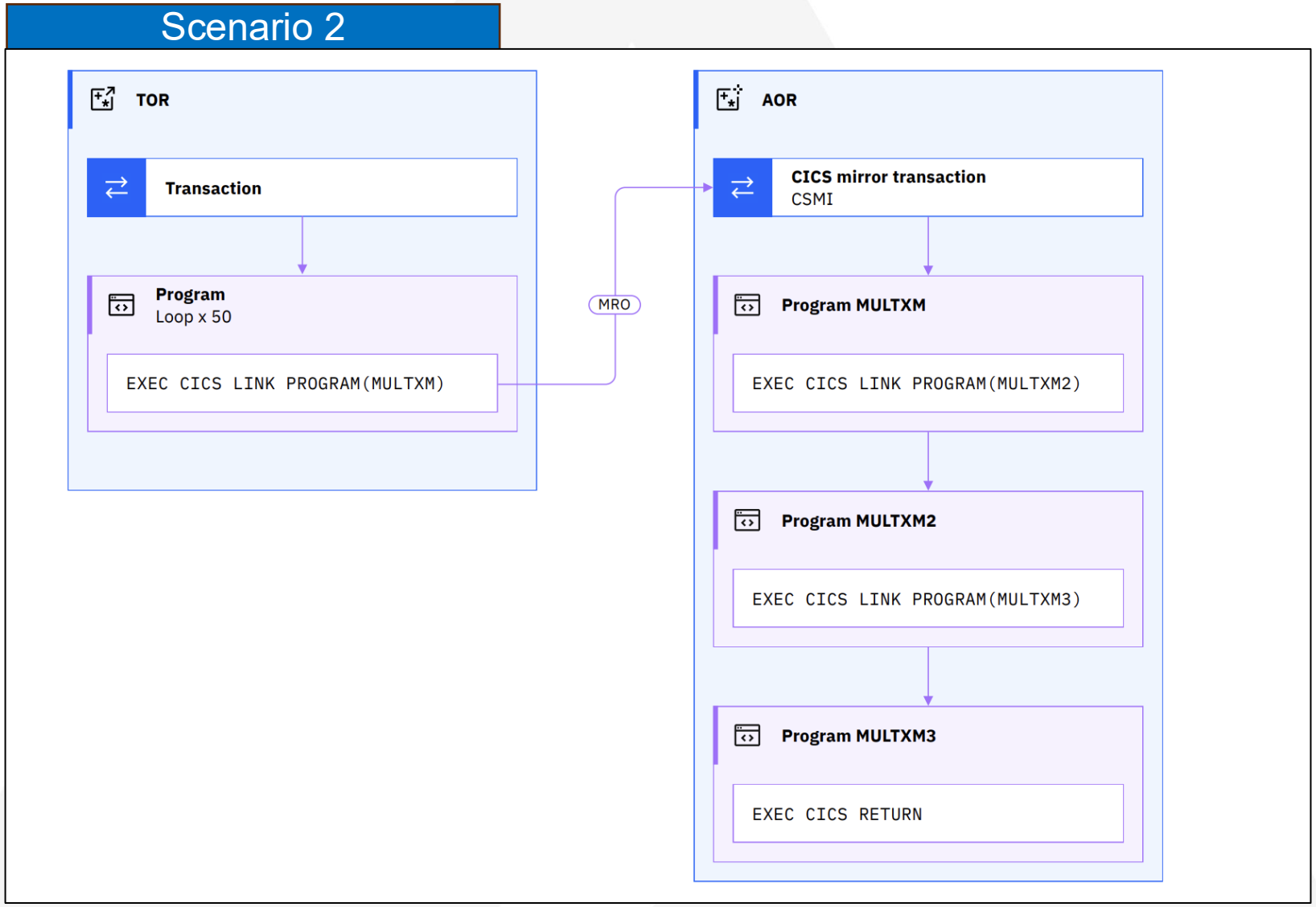
Test 2: Running workload with Security Discovery enabled

Test Case	Txn p/s	TOR CPU (%)	AOR CPU (%)	Total CPU cost per txn (ms)
Test 1 (No SDD)	2501	7.15	14.10	0.085
Test 2 (SDD)	2500	7.21	15.16	0.089



Increase in around 4 microseconds of CPU per txn

Security Discovery



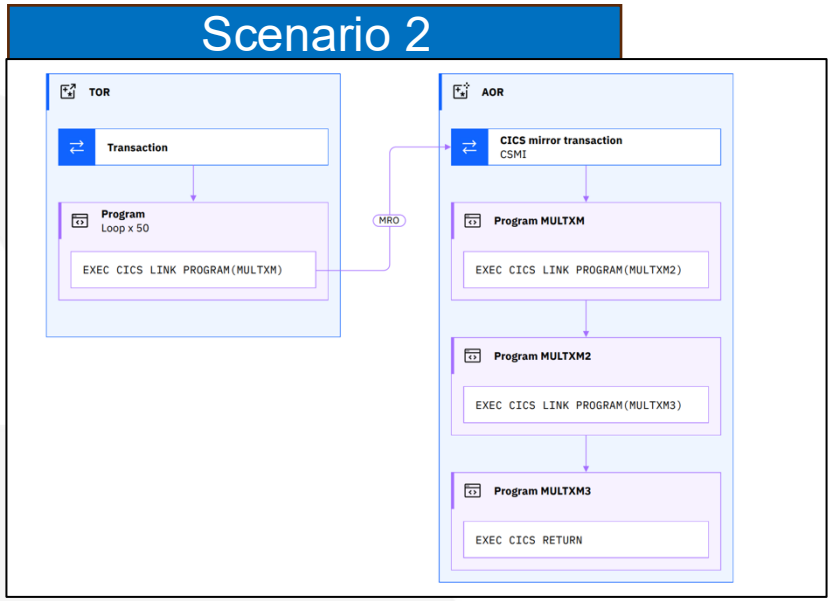
Security Discovery

Performance Testing

Test 1: Running workload with Security Discovery disabled

Test 2: Running workload with Security Discovery enabled

Test Case	Txn p/s	TOR CPU (%)	AOR CPU (%)	Total CPU cost per txn (ms)
Test 1 (No SDD)	1000	13.17	45.46	0.586
Test 2 (SDD)	1000	13.27	47.08	0.604



Increase in around 18 microseconds of CPU per txn

Security Discovery

Performance conclusion

There is an overhead in enabling Security Discovery

- Security Discovery is not permanent – short term cost
- Limit the overhead by fine tuning the configuration of discovery
- Generally, a small cost in the overall expense of enabling resource security

COST OF RESOURCE SECURITY

Cost of Resource Security

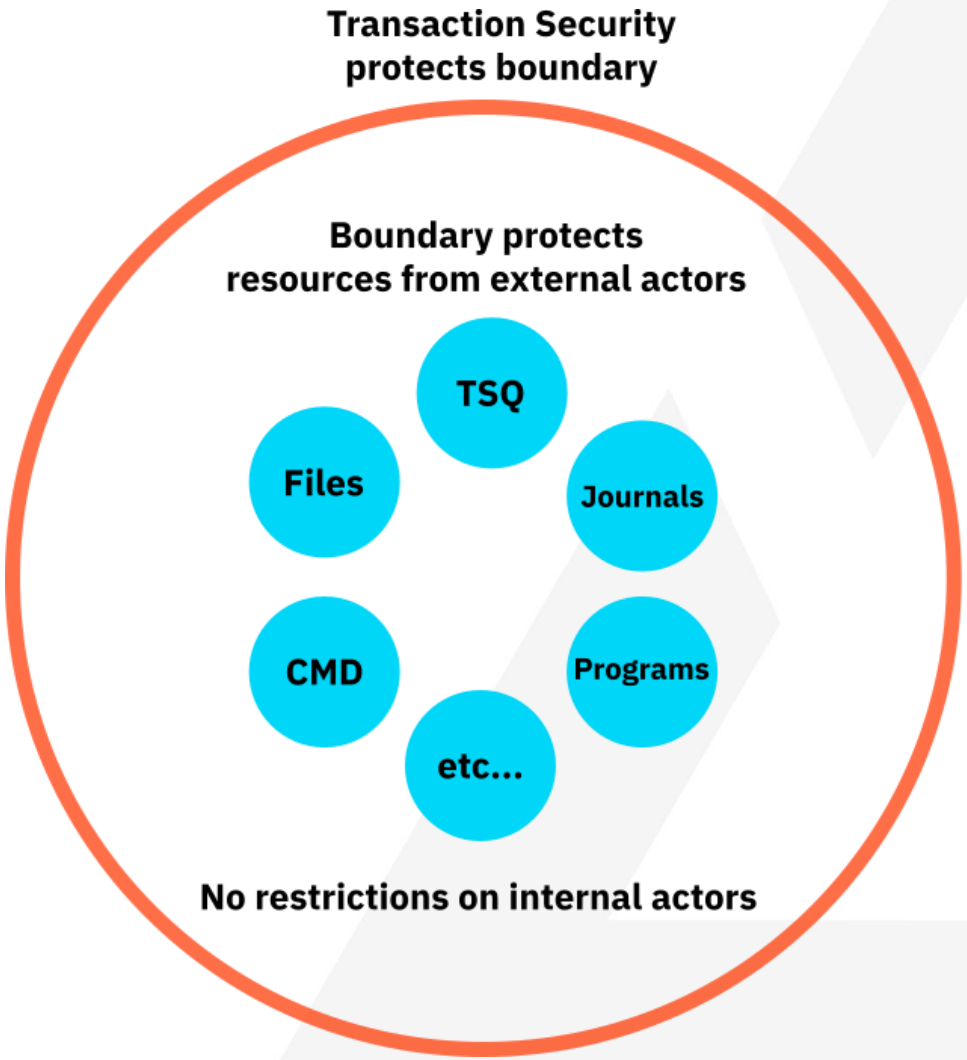
Overview

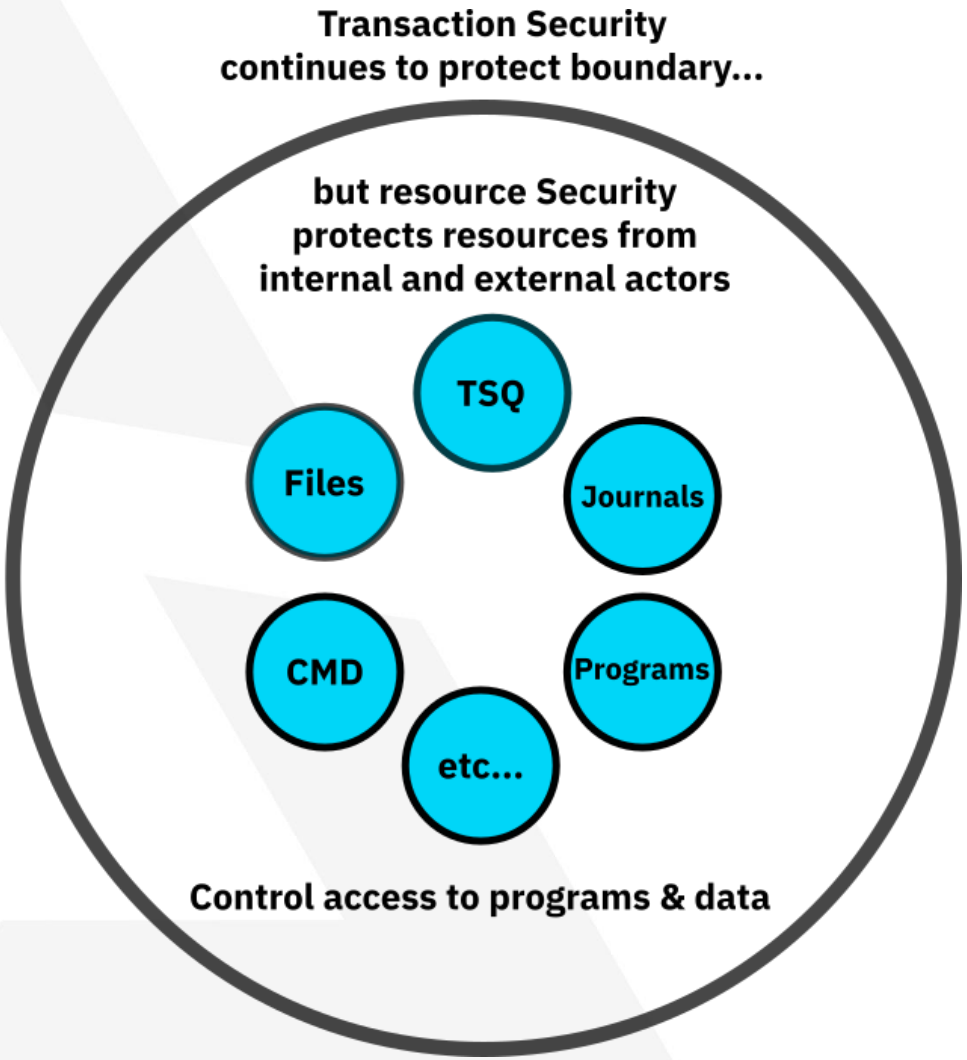
Newly defined TRANSACTION in CICS 6.2 `RESSEC(YES)` & `CMDSEC(YES)`

Enabling both command and resource security increases the number of security checks occurring in the entire region

- Customers pressured by regulatory standards and audit
- Concerns around the cost of increased checks
- How to best prepare for enabling

Cost of Resource Security





Cost of Resource Security

Performance Testing

Configuration

- RACF used as the ESM (External Security Manager)
- All RACF classes are RACLISTed
- SAF calls using FASTAUTH macro

Cost of Resource Security

Scenario 1

TOR → AOR:

- x10 looping using `EXEC CICS LINK PROGRAM`
- Writes to a TSQUEUE and deletes it

Test 1: RESSEC and CMDSEC set to NO (transaction security only)

Test 2: RESSEC and CMDSEC set to YES through SIT parameter.
XTST SIT set to YES with SECURITY(YES) on TSMODEL

Cost of Resource Security

Scenario 1

Test Case	Txn p/s	TOR CPU (%)	AOR CPU (%)	Total CPU cost per txn (ms)
Test 1 (No resource security)	2499	6.70	6.80	0.054
Test 2 (Resource security)	2500	7.10	14.89	0.088

Increase in around 34 microseconds of CPU per txn

Security calls per transaction = 22

Cost of Resource Security

Scenario 2 (DPL workload using 50x loop)

Test Case	Txn p/s	TOR CPU (%)	AOR CPU (%)	Total CPU cost per txn (ms)
Test 1 (No resource security)	1000	13.10	45.62	0.587
Test 2 (Resource security)	999	14.16	71.78	0.860

Increase in around 273 microseconds of CPU per txn

Security calls per transaction = 151

Cost of Resource Security

Scenario 3 (Basic DB2 workload)

Test Case	Txn p/s	Region CPU (%)	Total CPU cost per txn (ms)
Test 1 (No resource security)	1334	29.20	0.219
Test 2 (Resource security)	1333	29.97	0.225

Increase in around 6 microseconds of CPU per txn

Security calls per transaction = 3

Cost of Resource Security

Anticipating the cost

You can utilise the data within SECURITY and TRANSACTION MANAGER in CICS statistics

Number of security checks per task =

$$\frac{\text{(Successful resource authorization + Successful command authorization)}}{\text{(Total number of active user transactions)}}$$

Cost of Resource Security

Anticipating the cost

Once the number of security checks have been ascertained

- Divide the additional CPU cost between test 1 & 2 by the number of checks.
- This calculates the average cost per individual security check

This method can be used to calculate the cost of the additional checks region wide

Cost of Resource Security

```
TRANSACTION MANAGER STATISTICS
______________________________

  Total number of transactions (user + system)                  :                          749,691
  Current MAXTASKS limit                                        :                              900
  Time MAXTASKS last changed                                    :       02/09/2024  20:45:02.9238
  Current number of active user transactions                    :                                1
  Time last transaction attached                                :       02/09/2024  21:30:02.5895
  Current number of MAXTASK queued user transactions            :                                0
  Times the MAXTASKS limit reached                              :                                0
  Time the MAXTASKS limit last reached                          :       --/--/----  --:--:--.----
  Currently at MAXTASKS limit                                   :                               No
  Peak number of MAXTASK queued user transactions               :                                0
  Peak number of active user transactions                       :                               63
  Total number of active user transactions                      :                           749677
  Total number of MAXTASK delayed user transactions             :                                0
  Total MAXTASK queuing time                                    :                   000-00:00:00
  Total MAXTASK queuing time of currently queued user transactions  :                       00:00:00
```

Cost of Resource Security

```
SECURITY
________
  New ACEEs with ICRX                      :          0
  New ACEEs without ICRX                   :          0
  Current ACEEs with ICRX                  :          0
  Peak ACEEs with ICRX                     :          0
  Current ACEEs without ICRX               :       1002
  Peak ACEEs without ICRX                  :       1002
  Successful fastpath authentications      :          0
  Successful fullpath authentications      :          0
  Failed fullpath authentications          :          0
  Successful Kerberos authentications      :          0
  Failed Kerberos authentications          :          0
  Successful JWT creations                 :          0
  Failed JWT creations                     :          0
  Successful JWT authentications           :          0
  Failed JWT authentications               :          0
  Successful resource authorizations       :   16492875
  Failed resource authorizations           :          0
  Successful command authorizations        :          3
  Failed command authorizations            :          0
  Successful surrogate authorizations      :          0
  Failed surrogate authorizations          :          0
  Successful non-CICS authorizations       :          0
  Failed non-CICS authorizations           :          0
  Failed authorizations NOLOG NOTAUTH      :          0
  Failed authorizations NOLOG NOTFND       :          0
  Maximum parallel ESM requests            :        300
  Current parallel ESM requests            :          0
  Peak parallel ESM requests               :          0
  Maximum waiting ESM requests             :       9999
  Current waiting ESM requests             :          0
  Peak waiting ESM requests                :          0
```

Cost of Resource Security

Performance Conclusion

There is a sizable overhead in enabling resource and command security

- Subjective to the type of work being performed
- Effected by RACROUTE macro type (AUTH, FASTAUTH, etc)

Extra security isn't free

- You can be selective with which types of resource checks occur with SIT parms

XPPT DPLONLY

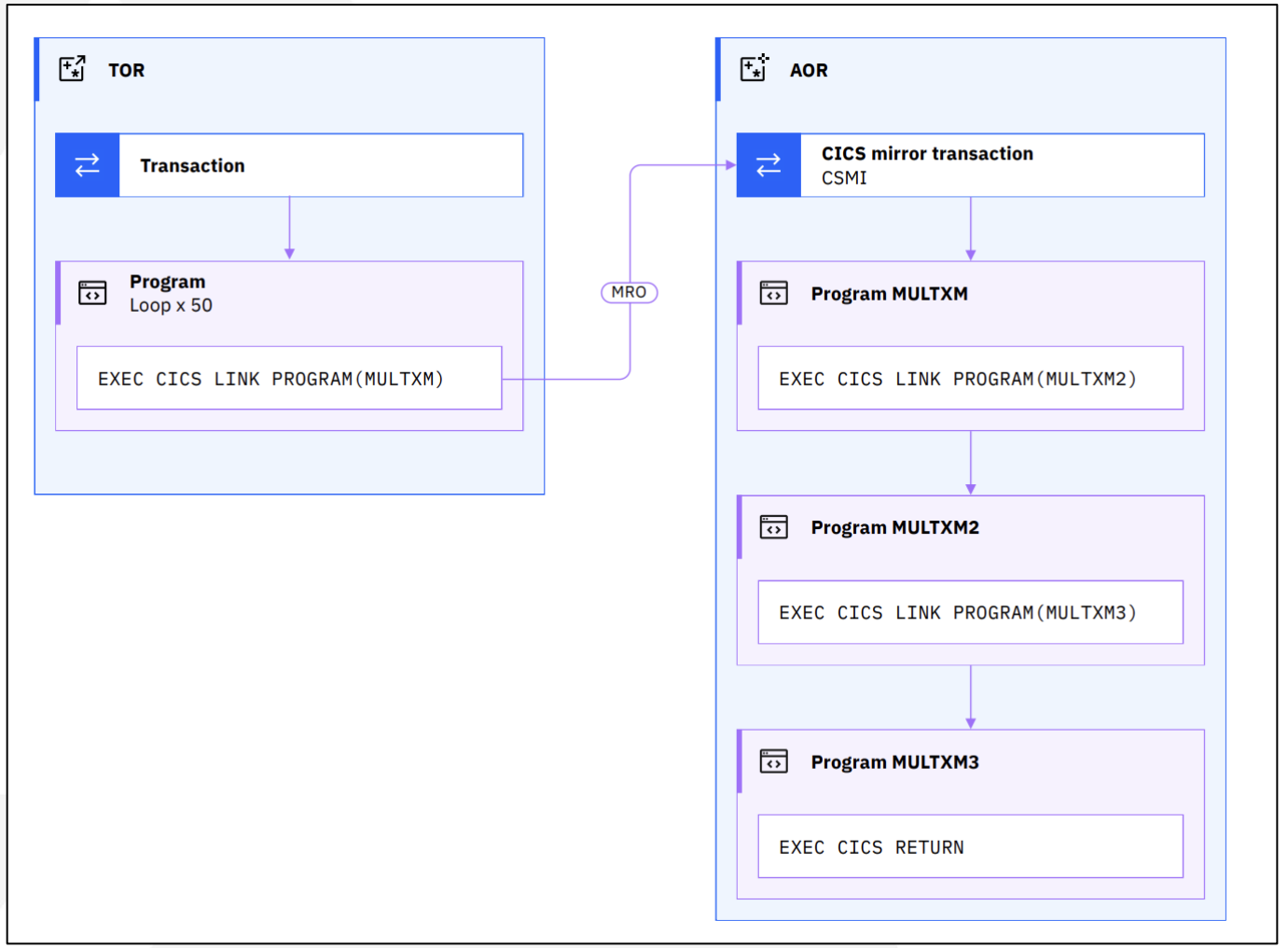
CICS TS 6.2 introduced DPLONLY for XPPT

To be selective on program security, DPLONLY enables the first program linked by CSMI to be checked only.

- Limits the amount of security checks whilst authenticating on initial program

XPPT DPLONLY

Test Case	Number of security authorizations in AOR
Test 1: XPPT=NO	299,927
Test 2: XPPT=(YES,ALL)	45,231,977
Test 3: XPPT=(YES,DPLONLY)	15,289,208



TLS ENHANCEMENTS

TLS Enhancements

Performance

CICS TS 6 now supports TLS 1.3

Workload driven consisting of a COBOL application:

- EXEC CICS WEB RECEIVE TOCONTAINER
- EXEC CICS PUT CONTAINER
- EXEC CICS WEB SEND CONTAINER

TLS 1.2 Cipher suite - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (C030) TLS 1.3 Cipher suite - TLS_AES_256_GCM_SHA384 (1302)

TLS Enhancements

Persistent Connections

Test Case	Requests p/ sec	Combined server region CPU (%)	Server TCPIP (%)	Server ICSF CPU (%)	Total CPU cost per request (%)
TLS 1.2	1000	18.00	2.10	0	0.201
TLS 1.3	1000	18.30	2.10	0	0.204
ATTLS TLS 1.2	1000	9.80	3.20	0	0.130
ATTLS TLS 1.3	1000	10.03	3.20	0	0.132

TLS 1.2 and 1.3 perform comparably when configured in CICS

Saving could be made by using AT-TLS handing off to System SSL – 72 microseconds of CPU

TLS Enhancements

Non-persistent Connections

Test Case	Requests p/ sec	Combined server region CPU (%)	Server TCPIP (%)	Server ICSF CPU (%)	Total CPU cost per request (%)
TLS 1.2	1000	43.67	4.97	0.4	0.490
TLS 1.3	1000	50.31	4.47	0.4	0.552
ATTLS TLS 1.2	1000	16.20	37.60	0.4	0.542
ATTLS TLS 1.3	1000	16.37	38.37	0.6	0.553

TLS 1.2 and 1.3 perform comparably when configured in CICS

Saving could be made by using AT-TLS handing off to System SSL – 72 microseconds of CPU

TLS Enhancements

Performance Conclusion

- AT-TLS delivers substantial CPU savings compared with direct CICS-to-System-SSL integration, reducing CICS-region processing cost.
- Savings arise largely because CICS avoids S8
  TCB switches when TLS processing is offloaded to AT-TLS.
- AT-TLS centralises configuration, making it easier to manage multiple secure ports and connections in a single policy file.
- AT-TLS should be strongly considered where possible due to both the performance benefits and operational simplicity.
- Workload characteristics vary between environments; customer systems may observe different CPU profiles depending on user volume, security setup, and operational complexity.

6.2: Cost of resource and command security

Last Updated: 2026-01-16 AI

For newly defined TRANSACTION resources in CICS® TS 6.2 both CMDSEC and RESSEC default to 'YES'.
This section describes the measurements that were carried out to evaluate the performance impact from enabling Resource and Command Security (**RESSEC=YES**, **CMDSEC=YES**). For more information, see Resource security.

Three independent workloads were used to collect measurements, as follows.

Infrastructure details

All CICS regions were running on an IBM z16® model A01, which was configured with three dedicated CPs for an equivalent server type of 3931-703. An IBM® Storage DS8950F unit was used to provide external storage. The CICS TS 6.2 environment was the code available at the general availability (GA) date of June 2024.

In all following scenarios, these conditions apply:
- External Security Manager (ESM) is RACF®
- The RACF classes that are used are RACLISTed
- CICS uses the RACROUTE REQUEST=FASTAUTH macro to call RACF, see CICS security control points

The transactions are defined by using the attributes that are shown in the example. The example shows that the default values and set both CMDSEC and RESSEC are overridden to 'no':

```
DEFINE TRANSACTION(xxx) GROUP(xxx)
PROGRAM(TERM)
TASKDATAKEY(USER)
TASKDATALOC(ANY)
TRANCLASS(DFHTCL00) RESSEC(NO) CMDSEC(NO)
```

Your feedback is important!

SHARE
EDUCATE • NETWORK • INFLUENCE

Submit a session evaluation for each session you attend:

www.share.org/evaluation