SDSF Security – How does it work on z/OS 2.5?

Rob Scott
Rocket Software
rscott@rs.com

Agenda

Background

- Internal security and why it has been removed

Security Fundamentals

- SDSF and SDSFAUX server requirements
- Connecting to SDSF

Using SDSF Panels

- Accessing panels
- Actions and overtypes

Controlling the scope of data

Advanced Functions

- Destination operator authority
- Foreign address space data access
- Issuing operator commands

Using SDSF to Understand SDSF Security

Background

SDSF historically had internal security based around ISFPARMS assembler source and load module or statements in ISFPRMxx member of PARMLIB

SAF security introduced to SDSF in early 1990s

- IBM have been strongly advising customers to migrate to SDSF SAF security for at least 25 years (!)

SDSF 2.4 and below performed security checks against SAF and if “no decision” a fallback to ISFPRMxx or ISFPARMS specification was performed.

SDSF 2.5 onwards all security checks are now SAF and performed by the SDSF server address space on behalf of the user

- Provides single security administration and audit functions on industry standard ESM software
- Server loads ISFPARMS as emergency fallback in case initial ISFPRMxx fails to activate
- Fallback to client ISFPARMS no longer taken
- Users require access to SERVER.NOPARM in SDSF class to use product if emergency fallback is in effect

Security Fundamentals

Security Fundamentals

SAF security required starting from z/OS 2.5

- SDSF Security Migration Guide (SC27-4942)
  - Plan your migration carefully – there are some provided tools but they are only **starting points**
  - Iterative process with **extensive** testing and validation required
  - Difficulty rises exponentially if JESSPOOL class not already active
  - APAR PH49811 adds new ISFNTCNV tool to aid conversion of NTBL/NTBLENT statements from ISFPRMxx
- SDSF Operation And Customization (SA23-2274)

SAF class SDSF used to protect SDSF resources and product functionality

- Ability to display certain panels
- Ability to take actions against objects shown on panels

SDSF will perform SAF checks for other classes such as JESSPOOL and OPERCMDS

- SDSF does not own the resources for these classes and the authority check is performed to improve the messages and/or displays presented to the user
  - There is also valid-add functionality of extra SAF checks for JESSPOOL for cancel/purge output actions
- If SAF authority is granted within SDSF code, the request is forwarded to the owning component (eg z/OS BCP or JES2) and they will perform their own SAF checks

SDSF Server Address Spaces

SDSF server mandatory from z/OS 2.5

- Security checks performed by SDSF server on behalf of user
- Unless installing maintenance, there is no need to restart SDSF server after IPL

SDSFAUX required for data collection and sysplex communication

Entries required in the STARTED class for each server address space

- OMVS segment required for SDSFAUX userid

SDSFAUX userid requires

- READ access to FACILITY class resource MVSADMIN.WLM.POLICY
- READ access to FACILITY class resource ERBSDS.MON2DATA

Connecting To SDSF

Each SDSF user must connect to the SDSF server

- One connection per task (TCB)

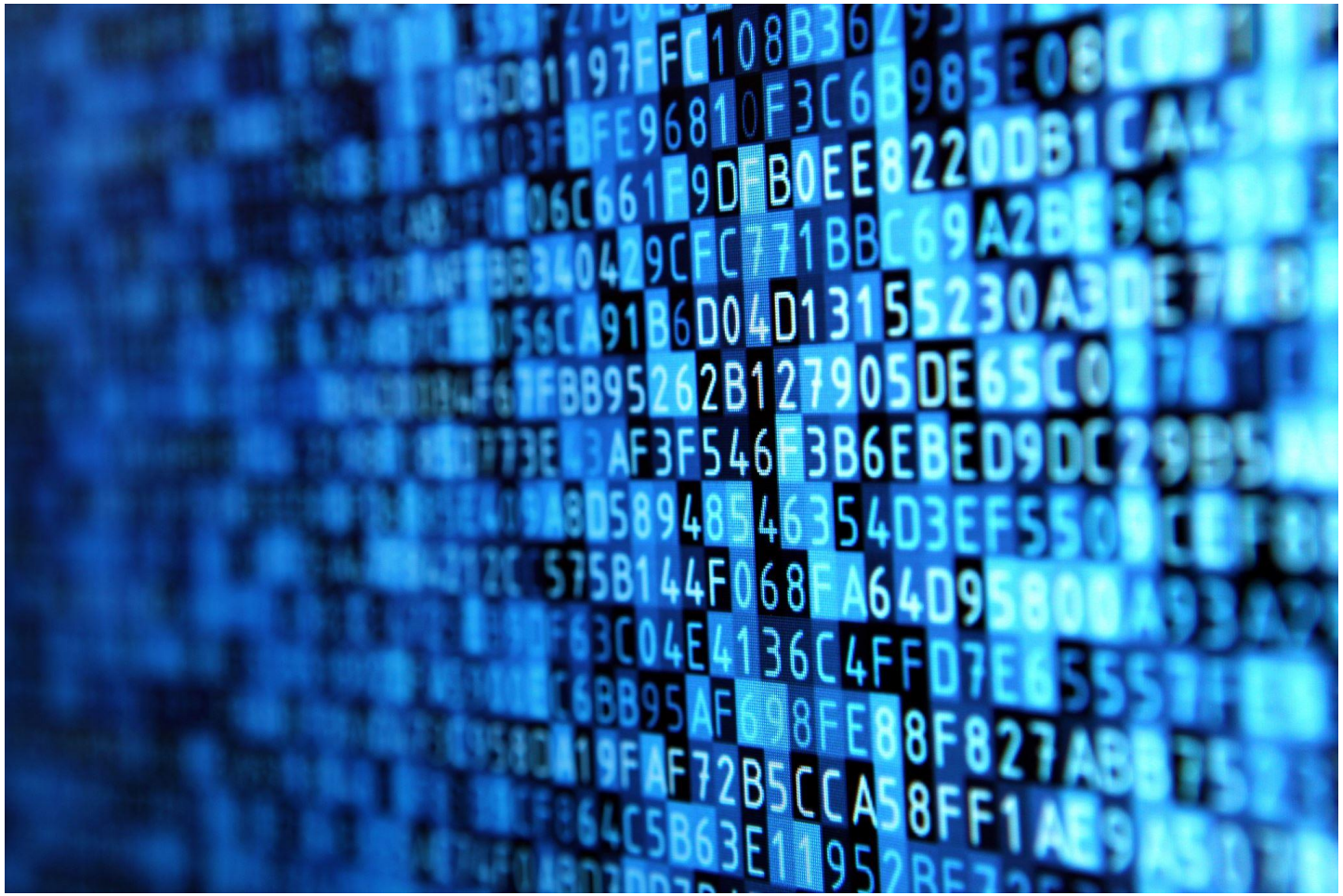
Ability to connect to SDSF requires

- READ access to SDSF class profile ISF.CONNECT.*sysname*

SDSF server attempts to place user into a SDSF group

- Groups are specified in ISFPRMxx PARMLIB member
- READ access to SDSF class profile GROUP.*name*.*sysname*
- User placed into first valid group in the sequence specified in ISFPRMxx
- If no matching group found the connection will fail
- SDSF groups dictate options and customizations
  - No longer used for authorization settings

Using SDSF Panels



SDSF Users

Three types of SDSF user in provided sample ISFPRMxx PARMLIB members in SISFJCL

- Systems programmer
- Operator
- General user – for example, a COBOL programmer

ISFPRM00

- Sufficient for most installations

ISFPRM01

- All statements provided with default values

Access To SDSF Panels

General user panels protected by

- READ access to SDSF class profile ISFCMD.DSP.*name.qualifier*
  - Think of “DSP” standing for “display”
- Includes active jobs and JES input and output queues

Systems programmer and operator panels protected by

- READ access to SDSF class profile ISFCMD.ODSP.*name.qualifier*
  - Think of “ODSP” standing for “operator display”
- Includes more system level displays and functions
- General users can be easily denied access to advanced displays by having no access to ISFCMD.ODSP.*.*

SDSF main panel will normally only show entries that user is authorized to

- General users have been seeing the same 5 to 8 choices for many years
- Over 60 other panels for systems programmers and operators
- SET MENU ALL shows entries for unauthorized and not applicable choices
- Display pull down on action bar shows all panels

Actions Typed In The “NP” Column

Access checked against resource that protects the row

- READ access for display style actions
- Higher access (UPDATE, CONTROL or ALTER) for change or delete style actions
- When action just displays another SDSF panel, then row object access is not required

JESSPOOL class profile used for active jobs and output

- Rows on ST, DA, H, I and O

Other IBM classes used if applicable

- For example, XFACILIT for CK rows (HealthCheck)

SDSF class profile used for other panel objects when generating operator commands

- ISF*xxxxx*.*qualifier_1*.*qualifier_2*…*qualifier_n*
- xxxxx = short name for panel object, e.g. “APF” for entries on APF panel
- Qualifiers depend on panel, e.g. the dataset name on APF

OPERCMDS profile checked when action generates z/OS operator command

- Checked after successful access to profile covering row object

Issuing Action Character – Example #1

User FRED types “H” to display the held output queue for the JES subsystem “JES2”

User “FRED” then issues “P” against JES jobid JOB01234 “BOWLING” on node “BEDROCK”. The job is owned by user “BARNEY”.

- “P” action requires ALTER access as “delete” style request.
- Row is protected by JESSPOOL profile
- OPERCMDS profile checked as “P” generates operator command
- Sequence of SAF checks :
  - READ access for SDSF profile ISFCMD.DSP.HELD.JES2
  - ALTER access for JESSPOOL profile BEDROCK.BARNEY.BOWLING.JOB01234.qualifiers
  - UPDATE access for OPERCMDS profile JES2.CANCEL.BATOUT

Issuing Action Character – Example #2

User FRED types “NA” on system WILMA to show the network activity and then types “DN” beside the row for TN3270 to display the connections.

- “DN” action requires READ as a “display” style request
- Row is protected by SDSF class profile ISFNETACT.*qualifiers*
- OPERCMDS profile checked as “DN” generated operator command
- Sequence of SAF checks
  - READ access for SDSF profile ISFCMD.ODSP.NETACT.WILMA
  - READ access for SDSF profile ISFNETACT.TN3270
  - READ access for OPERCMDS profile MVS.DISPLAY.TCPIP

Issuing Action Character – Example #3

User FRED types “CK” on system WILMA to show the HealthChecks and then types “DS” against the ASM_LOCAL_SLOT_USAGE check to display its status

- “DS” action requires READ as a “display” style request
- Row is protected by XFACILIT class profile HZS.qualifiers
- OPERCMDS profile checked as “DS” generates operator command
- Sequence of SAF checks
  - READ access for SDSF profile ISFCMD.ODSP.HCHECKER.WILMA
  - READ access for XFACILIT profile HZS.WILMA.IBMASM.ASM_LOCAL_SLOT_USAGE.QUERY
  - UPDATE access for OPERCMDS profile MVS.MODIFY.STC.HZSPROC.HZSPROC

Overtyping Values On Rows

Overtype columns protected by SDSF class profile

- ISFATTR.*type*.*column*
- *Type* describes the panel row object
- *Column* is the internal column name (not the title)
  - See “COLH” display for panel to see list of internal column names
- UPDATE access required

SAF check performed when column becomes first becomes visible in the display

- All of the column (both left and right bounds) must be visible on the screen
- If access allowed, the screen attributes for the column are changed to allow input
- If access denied, the screen attributes for the column are left as output only

If valid overtype value is detected, security processing matches the sequence described for “NP” actions

Controlling The Scope Of Data Displayed

Jobname PREFIX

PREFIX command

- Allows filtering by jobname using masking characters
  - Applies to active jobs and JES queue displays
- Only user with authority to PREFIX can change the default specified in the SDSF group they are assigned to
- READ access to SDSF class profile ISFCMD.FILTER.PREFIX

Jobname OWNER

OWNER command

- Allows filtering by job owner using masking characters
  - Applies to active jobs and JES queue displays
- Only user with authority to OWNER can change the default specified in the SDSF group they are assigned to
- READ access to SDSF class profile ISFCMD.FILTER.OWNER

Output Destination

DEST command

- Allows filtering by destination using name values
  - Applies to JES queue displays
- Only user with authority to DEST can change the defaults specified in the SDSF group they are assigned to
- READ access to SDSF class profile ISFCMD.FILTER.DEST

Enforcing Data Scope

Careful specification of PREFIX, OWNER and DEST can safeguard general users from seeing or manipulating jobs outside of their normal responsibilities

- SET DISPLAY ON causes values to be shown above the column titles
- No SAF authority for DEST, PREFIX and OWNER commands will restrict user to stay within values dictated by their GROUP in ISFPRMxx

ISFPRMxx Keywords

- IDEST Points to list of initial destinations
- OWNER NONE/USERID
- PREFIX NONE/USERID/GROUP

Grouping Jobname Output

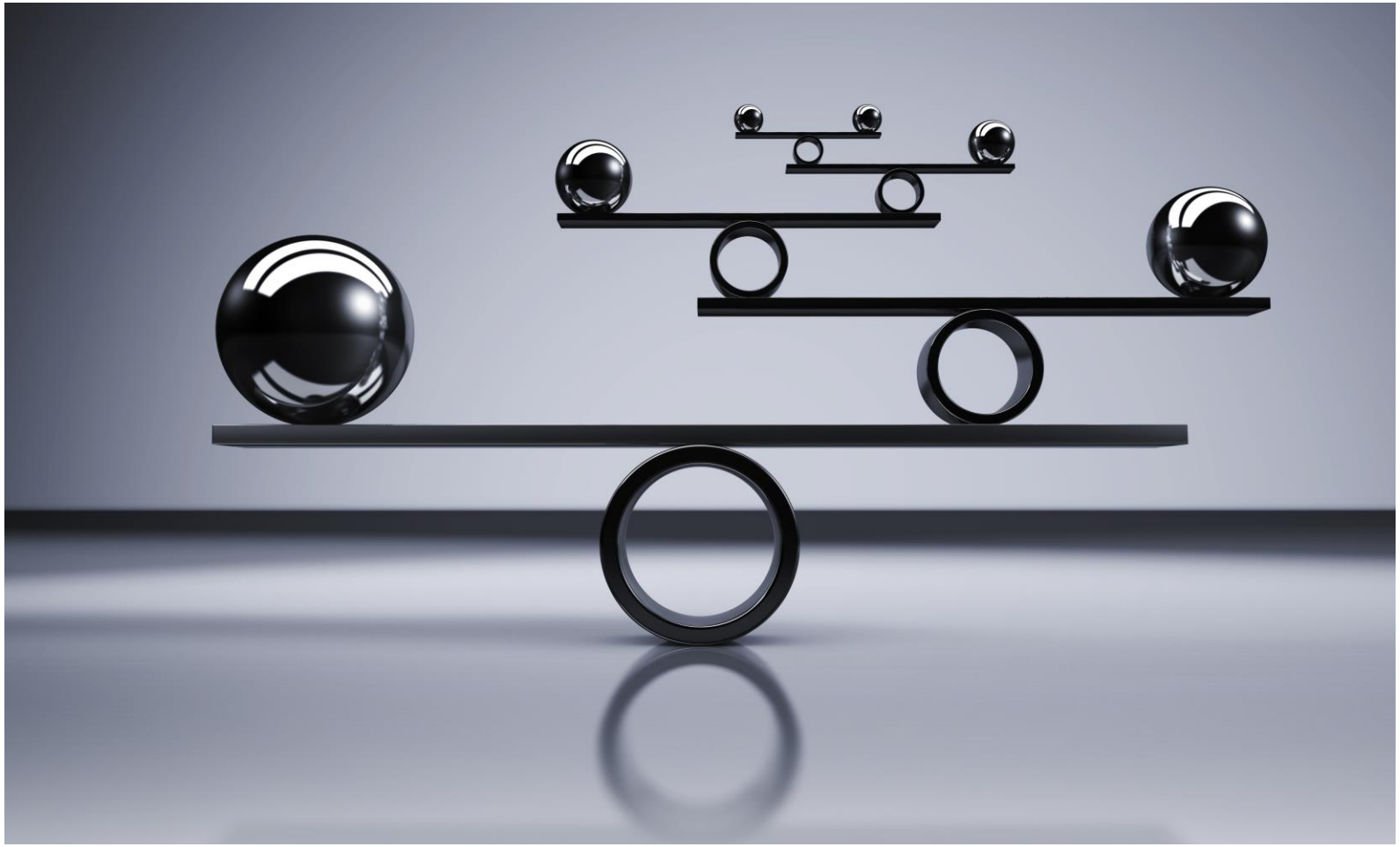
ISFPRMxx NTBL/NTBLENT statements allow specification of a “name table” using substring masking

- Coupled with IDSP or ICMD keywords on SDSF GROUP statements enabled jobs/userids to be grouped by masks for authority checking

SAF-only equivalent can be achieved by using RACF variables

- RACFVARS class profile specifies the RACF variable name – “&” followed by up to 7 characters – eg “&ISFJOB1” or “&ISFUID1”)
- Use RALTER ADDMEM commands to add entries to the RACFVARS profile
  - Example : RALTER RACFVARS &ISFJOB1 ADDMEM(*CICS*)
- JESSPOOL profile can use RACF variable as a qualifier
  - Example : N1.&ISFUID1.&ISFJOB1.*.**

Advanced Functions



Cross-System Data Access

SYSNAME command

- Allows the user to access data from one or more systems in the sysplex
- Masking characters can be used against the z/OS system name (CVTSNAME)
- Causes SDSF to send a message from the local SDSFAUX server to other SDSFAUX address spaces in the sysplex
- Responses are collected from all responding systems and presented locally to the user
- READ access to SDSF class profile ISFCMD.FILTER.SYSNAME
- If SYSNAME authority denied, then data gathering request is confined to the local system

Destination Operator Authority

Sometimes impractical to permit operators to every JESSPOOL class profile so that they can view or manage output

SDSF provides ability to define users with authority based on the destination for output rather than owner, jobname etc

- SDSF uses term “destination operator”

To be a destination operator, the userid must have

- READ access to SDSF class profile ISFOPER.DEST.*jesname*
- ALTER access to SDSF class profile ISFAUTH.DEST.*destname* for full management
  - READ access for just browse authority to output

Destination operator authority functions

- SDSF populates the BTOKRCID (network receiver userid) in the JES browse token
- SAF authority check by JES for JESSPOOL uses RECVR keyword on RACROUTE
- Log string added for auditing purposes

Foreign Address Space Data Access

Some SDSF commands require cross-memory access to other address spaces that are not necessarily under SDSF (or user) control

- Defines the term “foreign address space”
- Data is typically gathered by scheduling SRB into foreign ASID

Ability to gather foreign address space data protected by SDSF class profiles

- ISFJOB.*type.owner.jobname.sysname*
- READ access required
- If no owner assigned to address space then SDSF uses value “++++++++”
- Type values
  - DDNAME JDD action
  - MODULE JC action
  - TASK JT action
  - STORAGE JM, JMO and MEM

Restricting Operator Command Origin To SDSF

Actions in SDSF may result in operator command being generated

Sometimes not desirable to allow user ability to issue equivalent operator commands outside of SDSF control

Solution is to specify “WHEN(CONSOLE(SDSF))” on the PERMIT statement for the OPERCMDS class profile

- PERMIT JES2.CANCEL.BATOUT CLASS(OPERCMDS) ID(FRED) ACC(UPDATE) WHEN(CONSOLE(SDSF))

Requires CONSOLE class to be active and the “SDSF” console defined as a profile within the class.

Issuing Operator Commands

Freeform operator commands using “/”

- Or ISFSLASH function in REXX

READ access to SDSF class profile

- ISFOPER.SYSTEM
- Only restricts user from issuing operator commands in SDSF
  - Other authorized programs can issue operator commands outside of SDSF

Command text passed directly to z/OS to process

- No attempt made by SDSF to parse the freeform text and internally issue OPERCMDS authority checks
- WHEN(CONSOLE(SDSF)) does not apply
- z/OS BCP will issue SAF OPERCMDS check on the command passed to it

Handling “No Decision” Situations

SAF authority check return code

- RC=0 User access is permitted to the profile covering the resource
- RC=4 No decision can be made
- RC=8 User access is denied to the profile covering the resource

No decision circumstances

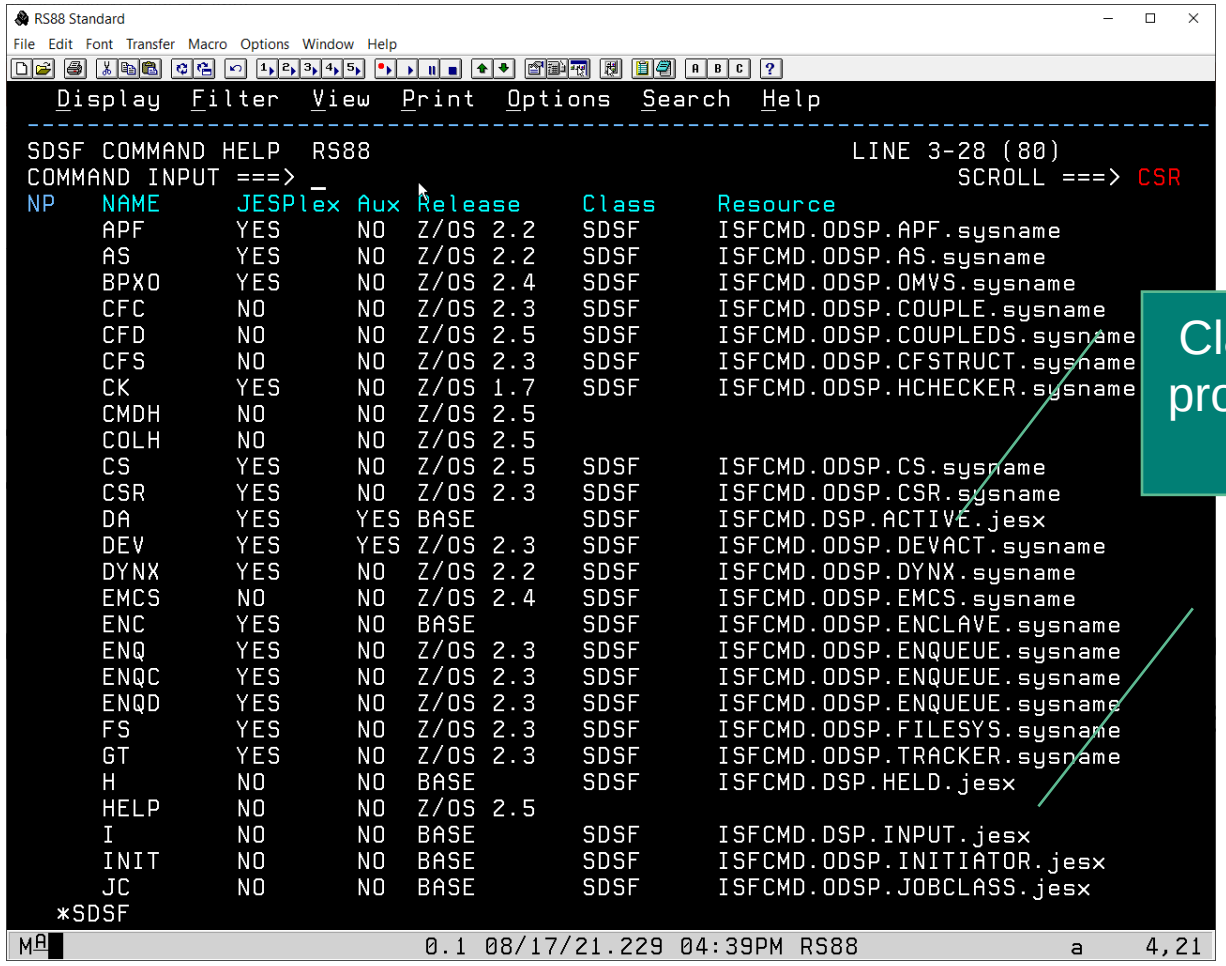
- Class is not active
- Class is active but not RACLISTed (applies to certain classes only)
- Class is active (and RACLISTed if applicable) but no matching profile found
  - No “catch-all” profile “**” defined with site-defined UACC value

ISFPRMxx CONNECT statement keyword AUXSAF(FAILRC4/NOFAILRC4)

- FAILRC4 Convert RC=4 from SAF into RC=8 (Default)
- NOFAILRC4 Convert RC=4 from SAF into RC=0

Using SDSF To Understand SDSF Security

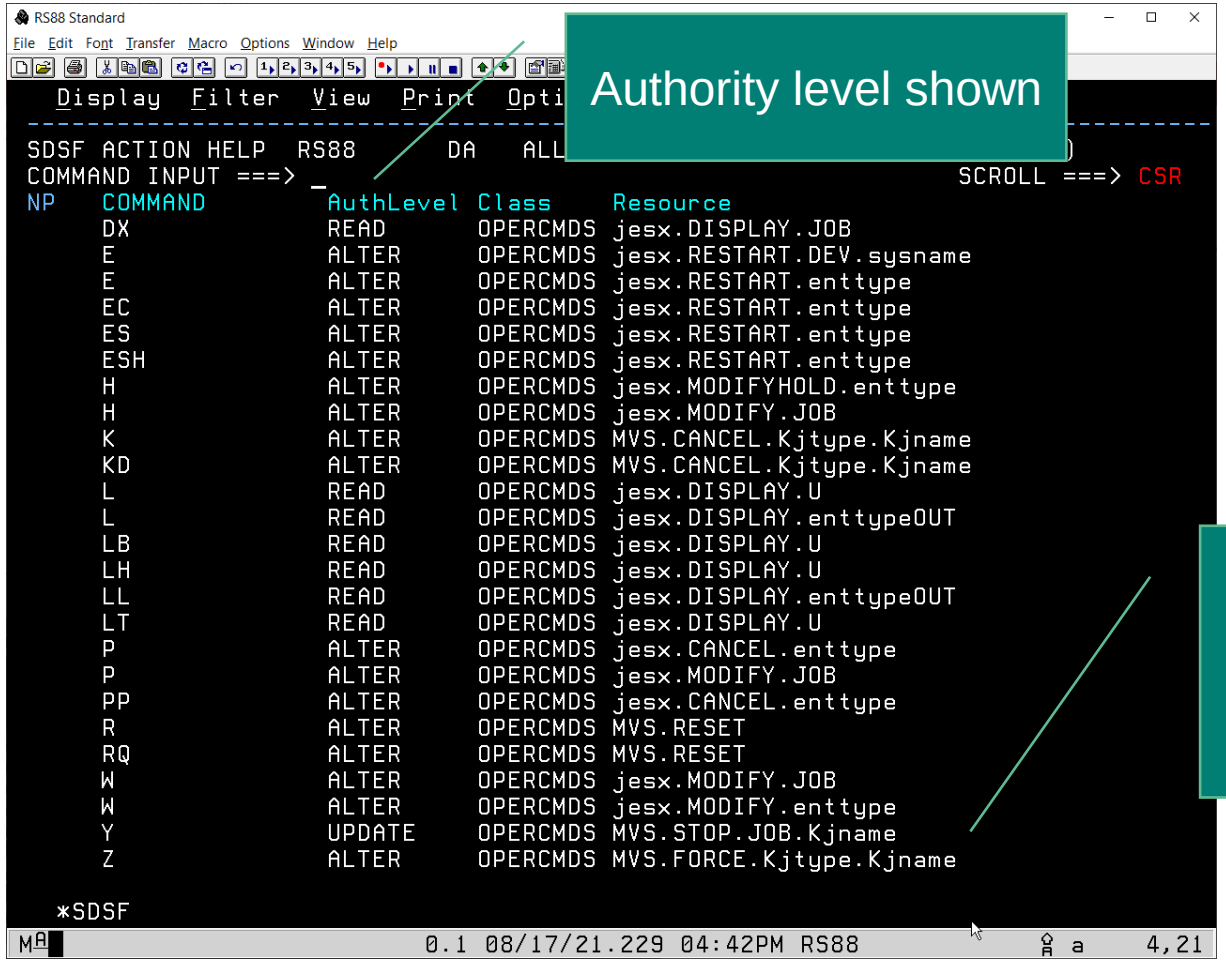
Primary Command Attributes - CMDH



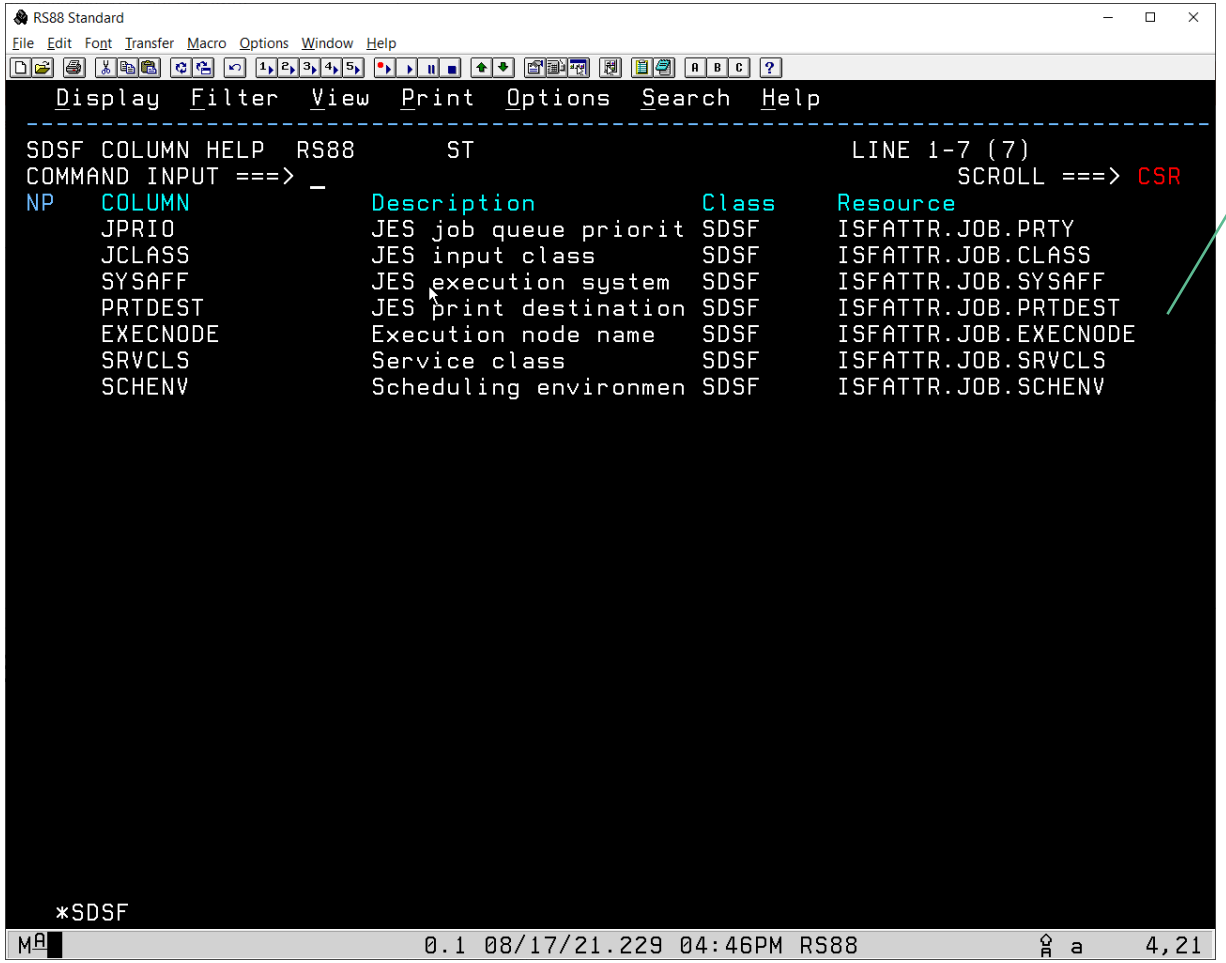
Class and resource protecting command shown

Note that basic user interface commands are not protected

Action Character Attributes - ACTH



Overtype Columns - COLH



Overtype columns show resource name and class.

Note that internal column name used and not the title.

Security Trace

Independent of normal SDSF trace

Can be used to see all SAF checks issued by SDSF

Trace output sent to ULOG by default

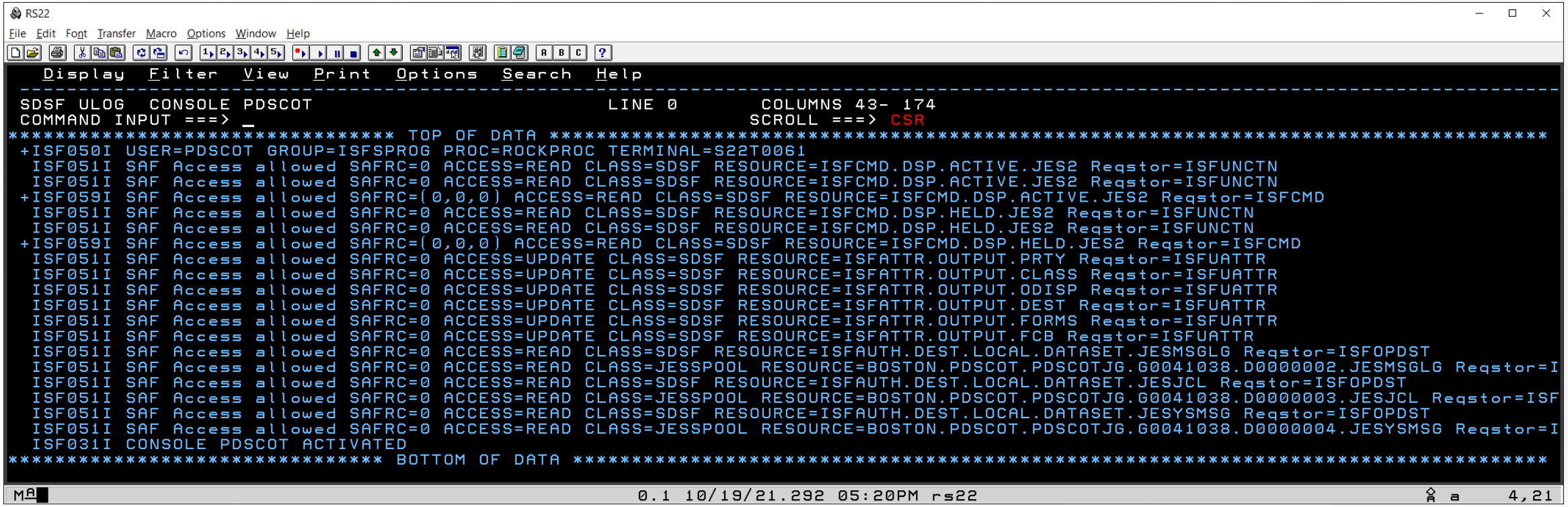
Issued via SET command

- SET SECTRACE ON
- SET SECTRACE OFF
- SET SECTRACE WTP
  - Issue “write to programmer” style WTOs instead of ULOG

Special DDNames

- Useful for tracing initialization
- ISFSECTR Enables SECTRACE ON
- ISFSECTW Enables SECTRACE WTP

Security Trace Output



SDSF Healthcheck

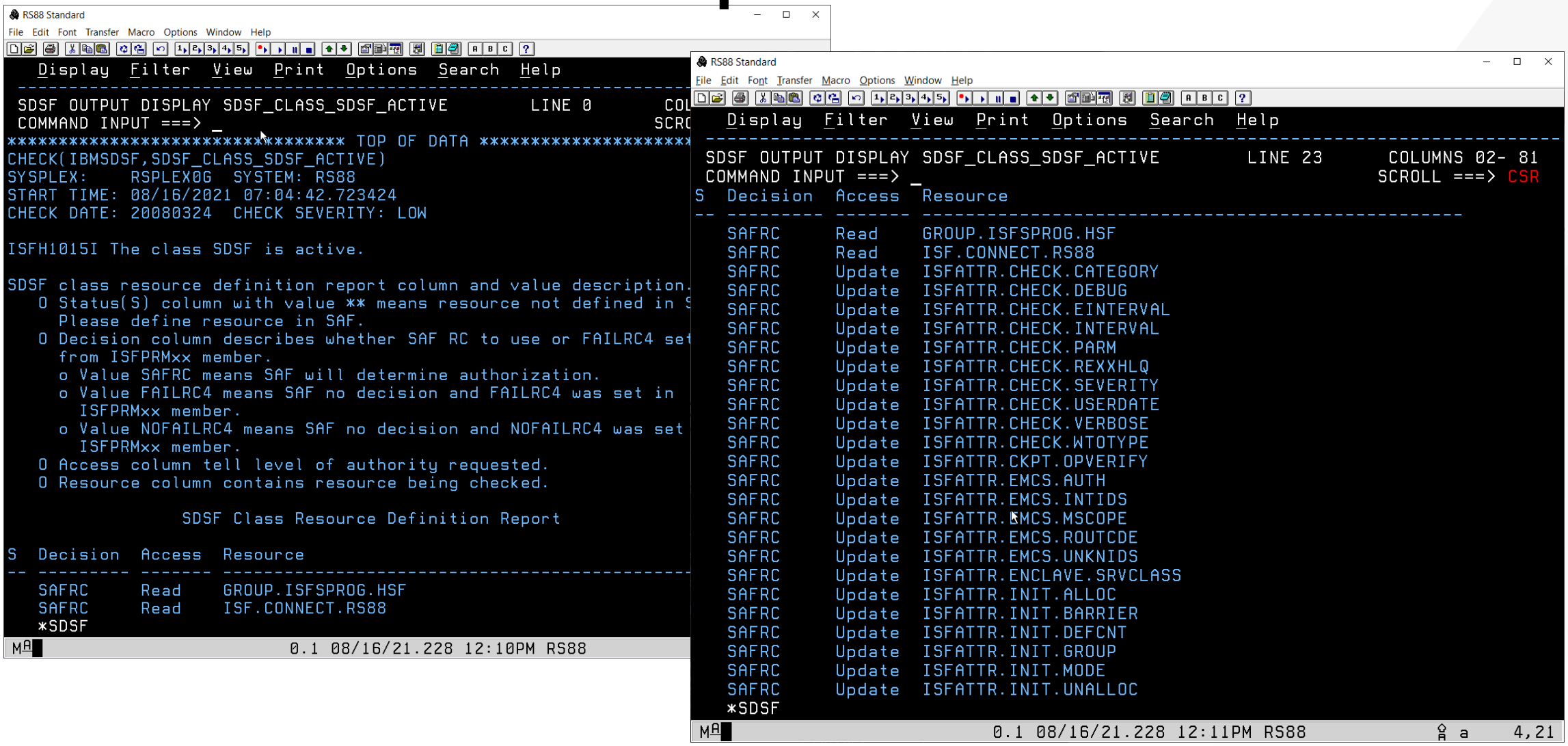
SDSF_CLASS_SDSF_ACTIVE

Verifies if SDSF class is active

Produces report on SDSF resources and how the security decision is derived

- SAFRC
  - Covering profile found and SAF return code will be used
- FAILRC4
  - No covering profile found and access will be denied due to AUXSAF(FAILRC4) in ISFPRMxx
- NOFAILRC4
  - No covering profile found and access will be allowed due to AUXSAF(NOFAILRC4) in ISFPRMxx
- Always wise to verify all entries that do NOT have SAFRC to ensure that you do not have exposures due to missing or incorrect profiles

SDSF Healthcheck Output



SDSF for z/OS 3.1 Enhancements

- New RACF themed displays
  - Classes
  - Profiles
  - Access lists
  - Options
  - Profile “browse”
- Allows the systems programmer to view RACF information easily
  - Uses R_admin callable service
  - Just “view” – no ability to change RACF environment

SDSF for z/OS 3.1 – RACF Screenshots