BaphCorp

SAFCHECK

User's Guide

Version 3.3

January 2023

Alain Barthélemy

Table of contents

Figures

News & Evolutions

The contributions of the different versions are as follows:

Version 1.1

- Original version.

Version 1.2

- Add default access.

Version 2.7

- Possibility to validate the rights of another user.

Version 3.3

- Possibility to validate DATASET class.
- Possibility to put the resource between cotes

How does SAFCHECK work

SAFCHECK is an assembler program allowing you to check whether a user has specific rights to a general RACF resource or to a Dataset. You get the result simply by a return code.
SAFCHECK is very useful for securing Rexx development under TSO and can be used in a batch chain to activate or deactivate processing

Using SAFCHECK

SAFCHECK is intended to be used either under TSO where it uses the Command Processor([1]), or in batch with EXEC PGM=SAFCHECK. Be careful, a batch TSO is considered a TSO as soon as it calls an IKJEFTxx program.
SAFCHECK requires at least two parameters: the resource class and the desired resource. If you want to test access greater than *Read*, you must indicate it in the third parameter. If you want to test the rights of another user, you must add the userid to test after the requested access.
The last two parameters take a default value if they are coded with *.
Note that it is possible to define the resource between cotes, but this is of no interest.

Implementations Constrtraints

- To work correctly, SAFCHECK requires to be placed in an APF loadlib.
- As SAFCHECK uses the TSO Command Processor, it must be definbed as authorized command in the IKJTSOxx member.

```
AUTHCMD NAMES(          /* AUTHORIZED COMMANDS     */ +
   . . . . .                                          +
    SAFCHECK                                          +
   . . . . .                                          +
```

Figure 1 : Definition in IKJTSOxx

Calling SAFCHECK in TSO

Below is an example of using SAFCHECK. Note that neutralizing WTPMSG prevents the display of a possible ESM error message on the screen.

```
/* Rexx : Check a right */
   Say "** Class Resource Access User :"
   Parse pull Classe Rsrc Acces User .
   Address TSO
   'SAFCHECK' Classe Rsrc Acces User
   Select
      When Rc = 0 then say "Resource authorized"
      When Rc = 4 then say "Resource unknown"
      When Rc = 8 then say "Resource forbidden"
      Otherwise say "Parameter error"
   End
Return 0
```

Figure 2 : SAFCHECK under TSO

[1] The SAFCHECK program must be defined in the AUTHCMD paragraph of IKJTSOxx

Calling SAFCHECK in Batch

Below is an example of using SAFCHECK in Batch followed by a call in TSO batch. The second step uses the TSO Command Processor as an interactive TSO call.

```
//SAFCHKBT JOB ('BaphCorp'),'Test Pgm',MSGCLASS=H,CLASS=A,
//             REGION=7M,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//*
//*---------------------*
//* SAFCHECK as program *
//*---------------------*
//*
//SAFCHKBT EXEC PGM=SAFCHECK,
//         PARM='FACILITY BPX.SUPERUSER read TzssY01'
//STEPLIB  DD  DISP=SHR,DSN=TZSSY03.USER.LOADLIB
//SYSPRINT DD  SYSOUT=*
//SYSUDUMP DD  SYSOUT=*
//*
//*---------------------------------*
//* SAFCHECK with command processor *
//*---------------------------------*
//*
//SAFCHKTS EXEC  PGM=IKJEFT1B,
//         PARM='%SAFCHECK FACILITY BPX.SUPERUSER read'
//STEPLIB  DD  DISP=SHR,DSN=TZSSY03.USER.LOADLIB
//SYSEXEC  DD  DISP=SHR,DSN=TZSSY03.USER.EXEC
//SYSTSPRT DD  SYSOUT=*
//SYSUDUMP DD  SYSOUT=*
//SYSTSIN  DD  DUMMY
//
```

Figure 3 : SAFCHECK in Batch

SAFCHECK syntax

SAFCHECK	***Class***	***Resource***	[***Access***	[***User***]]

The parameters ***Class***, ***Access*** and ***User*** may be written in either upper or lower case, only the ***Resource*** parameter is case sensitive.
The parameters ***Access*** and ***User*** are optional.

Description

Classe	Indicates the resource class
Resource	Indicates the resource to test (Be careful : case sensitive)
Access	Indicates the desired access The accepted values are : READ, UPDATE, CONTROL ou ALTER By indicating * we take the default value (**READ**)
User	Indicates the user for whom the right is tested. Specifying * amounts to using the user who calls the program.

Note :

- The resource can be defined between cotes, if it contains blanks they will be preserved and the result will be an *Abend 282-5C*

Return codes

0	Resource allowed.
4	Resource or class unknown.
8	Resource forbidden.
12	Parameter error.