



Introduction and Getting Started with the IBM Health Checker for z/OS

Marna Walle
IBM, Systems and Technology Group
Poughkeepsie, New York
mwalle@us.ibm.com

February 5, 2013
Session #13118



Trademarks

The following are trademarks of the International Business Machines Corporation in the United States, other countries, or both.

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business(logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Notice Regarding Specialty Engines (e.g., zIIPs, zAAPs and IFLs):

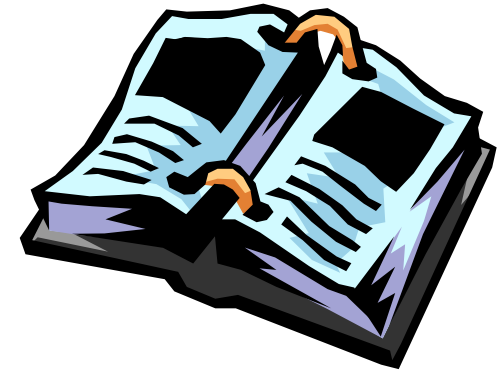
Any information contained in this document regarding Specialty Engines ("SEs") and SE eligible workloads provides only general descriptions of the types and portions of workloads that are eligible for execution on Specialty Engines (e.g., zIIPs, zAAPs, and IFLs). IBM authorizes customers to use IBM SE only to execute the processing of Eligible Workloads of specific Programs expressly authorized by IBM as specified in the "Authorized Use Table for IBM Machines" provided at www.ibm.com/systems/support/machine_warranties/machine_code/aut.html ("AUT").

No other workload processing is authorized for execution on an SE.

IBM offers SEs at a lower price than General Processors/Central Processors because customers are authorized to use SEs only to process certain types and/or amounts of workloads as specified by IBM in the AUT.

Introduction to IBM Health Checker for z/OS Agenda

- Why do we need a health checker?
- IBM Health Checker for z/OS concepts
- Setting up IBM Health Checker for z/OS
- Sample check outputs
- Check exceptions – what they look like and what to do about them
- Getting even more out of IBM Health Checker for z/OS:
 - Writing your own checks for IBM Health Checker for z/OS
- ***Advanced Topic: new items in z/OS R12 and R13!***
- Backup slides: “Demo” of setting up



Why do we need a health checker?



- **Analysis of outages showed:**
 - Significant number were avoidable: for example, bad configurations with single points of failure
 - Configurations that were less than optimal: for example, unnecessary performance bottlenecks
- **Situation exacerbated by:**
 - Parallel sysplex complex configuration requirements
 - Experienced skills are limited
 - Rare failures mean less experience by operations staff

Why do we need a health checker? ...continued

- **Many options for flexibility:**
 - Sometimes, default values are best guesses.
 - Best practices may not become known until good exposure in many environments
- **Best practices are not widely known or implemented:**
 - Many sources of best practices: product pubs, WSC Flashes, White Papers, wizards, ...
 - Hard to determine applicability
 - May be out of date
 - Just providing documentation has a limited affect



Why do we need a health checker? ...continued

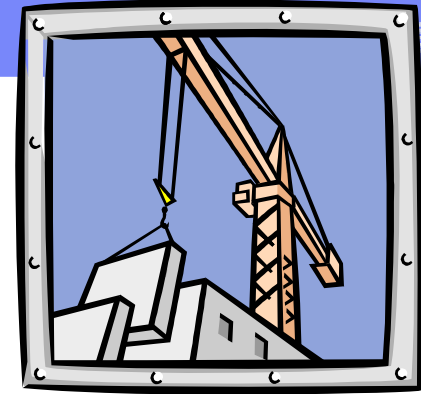
■ **Beyond Best Practices!**

- IBM Health Checker for z/OS checks (which are INACTIVE) can assist with migration action applicability determination.
 - The migration action is not performed by the check, but can alert you that you need to perform the migration action, then check that it was done correctly.
- Predictive Failure Analysis (PFA) surfaces reports via IBM Health Checker for z/OS.
 - PFA's intention is to convert "sick, but not dead" to a correctable incident.

- ***In a nutshell, using health checker can help you avoid outages, help with migration, and display your soft failures!***



IBM Health Checker for z/OS – The Pieces



- **Consists of two parts:**
 1. Framework – interface that allows you to run and manage checks. Shipped as part of z/OS BCP.
 2. Individual Checks – programs that examine specific settings or values for potential problems
 - Shipped and owned by the component, element, or product.
 - Checks are provided by some ISVs, and you can write your own!

A Health Checker instance can run on each system in the sysplex. There is only one instance of Health Checker on a single system.

“GLOBAL” checks run only on one system in a sysplex. This is to avoid running redundant copies of “Sysplex aware” checks. Checks of any locale can be declared “global”.

IBM Health Checker for z/OS – The Check Concepts



- **Check Values:** contains pre-defined values such as: interval, severity, and parameters.
 - Modify values via: SDSF, HZSPRMxx parmlib member, or MODIFY command
- **Check Output:** issued as messages. Exceptions also produce WTO messages.
 - Details can be viewed via: SDSF, HZSPRINT utility
- **Managing Checks:** print, display, activate/deactivate, add, refresh, run, update values temporarily or permanently
 - Permanent check updates can only be done via POLICY statements in HZSPRMxx

IBM Health Checker for z/OS – Types of Checks



- **Local**: runs in the IBM Health Checker for z/OS address space. Is authorized and can be written in assembler or Metal C.
- **Remote**: runs outside the IBM Health Checker for z/OS address space. May be authorized or non-authorized, and can be written in assembler or Metal C.
- **System REXX**: a kind of remote check. Runs in a System REXX address space. Is authorized (residing in an APF authorized library), and is written in System REXX.

*In theory, any language could be used that could call the required health checker services and conform to system linkage conventions.

IBM Health Checker for z/OS – Set up

Use the following steps to set up and start IBM Health Checker for z/OS:

1. Satisfy software requirements for IBM Health Checker for z/OS, as needed
2. Allocate the HZSPDATA data set to save check data between restarts
3. Define log streams to keep a record of the check output, as needed
4. Create security definitions
5. Set up customization and security for SDSF support for IBM Health Checker for z/OS in IBM Health Checker in *z/OS SDSF Operation and Customization* (*not required, but extremely helpful!*)
6. Create HZSPRMxx from the HZSPRM00 parmlib member
7. Start IBM Health Checker for z/OS
8. Set up the HZSPRINT utility, as needed
9. Obtain checks for IBM Health Checker for z/OS

You may skip grayed items, if you want a quick setup!



IBM Health Checker for z/OS – Set up – 1. Satisfy Software Requirements, as needed

Some IBM Health Checker for z/OS checks are written in **System REXX**.

- **Exploiters of System REXX (not just checks) require both of the following:**
 - **Compiled REXX** needs either Alternate Library for REXX (available in z/OS since V1R9) or the IBM Library for REXX on zSeries (5695-014). The runtime library will need to be available to the system:
 - REXX.SEAGALT can be put in the LNKLIST.
 - REXX.SEAGLPA can be put in the LPALST.
 - If you do not APF-authorize the library, you will get an abend when the runtime routines are requested by an APF-Authorized routine.
 - **System REXX** customization must be performed. Customization tasks are documented in "System REXX" in *z/OS MVS Programming: Authorized Assembler Services Guide*.

IBM Health Checker for z/OS – Set up – 2. Allocate HZSPDATA data set

Allocate the HZSPDATA data set to save check data between restarts

- Some checks use the HZSPDATA data set to save data required as part of their processing between restarts of the system or IBM Health Checker for z/OS.
- Use the HZSALLCP sample JCL from SYS1.SAMPLIB.
- Must have one HZSPDATA data set per system image! Therefore, recommended to use &SYSNAME. in the HZSPDATA name.

```
//HZSALLCP EXEC PGM=HZSAIEOF,REGION=4096K,TIME=1440
//HZSPDATA DD DSN=SYS1.system_name.HZSPDATA,DISP=(NEW,CATLG),
//          SPACE=(4096,(100,400)),UNIT=SYSDA,
//          DCB=(DSORG=PS,RECFM=FB,LRECL=4096)
//SYSPRINT DD DUMMY
```

Will be in HZSPROC

Don't change these attributes



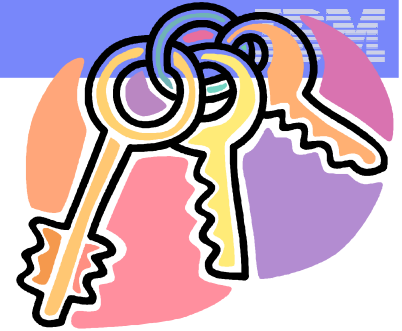
IBM Health Checker for z/OS – Set up – 3. Define log streams

Define log streams to keep a record of the check output, as needed

- Only the last iteration of a check is retained in the message buffer.
- It's a good idea to retain historical data about your check results. To do that, you must define and connect to a log stream (either CF or DASD-only).

Steps:

1. Plan for and set up the log streams. You may have one log stream per system or (when using CF logstreams) one log stream for multiple systems. Must begin with *HZS*.
2. Enable your log streams via `MODIFY` command or in `HZSPRMxx` parmlib member.
 - For permanent specification, you must use `HZSPRMxx` parmlib member.
3. You can disable the log stream with a `MODIFY` command.



IBM Health Checker for z/OS – Set up – 4. Create security definitions

Create security definitions

1. Set up security for the IBM Health Checker started task, similar to other started tasks:
 - a) Create user ID for IBM Health Checker for z/OS that has superuser authority. Connect that superuser ID to a group.
 - b) Associate that superuser ID to the started task, HZSPROC.
 - c) Give that superuser ID update access to HZSPDATA data set you allocated on each system.
 - d) Give that superuser ID read access to HZSPRMxx parmlib members.
 - e) If using log streams, give update access for that superuser ID to each log stream(s).
 - f) Some checks may require special authority for the Health Checker ID to access resources. That special authority should be told to you by the specific check.
2. If using HZSPRINT utility, follow instructions in the *User's Guide*.

RACF-supplied class, XFACILIT, is used.

As of z/OS R12, BPX.SUPERUSER access may be used instead of UID(0).

IBM Health Checker for z/OS – Set up – 5. Set up SDSF support

Set up customization and security for SDSF support, if desired.

This is very helpful, and probably easier than F HZSPROC or HZSPRINT only!

- SDSF provides support to make management of your checks easier with the CK panel for the IBM Health Checker for z/OS.
- You can use the CK panel to display checks, attributes, and status, taking advantage of standard SDSF sort, filter, and arrange support.
- Use “Protecting checks” in *z/OS SDSF Operation and Customization*:
 - Authorize users to the CK command using either the AUTH parameter of ISFPARMS or SAF.
 - Authorize users to action characters or overtypeable fields with ISFPARMS or SAF
 - Customize columns on the panel for groups of users with ISFPARMS.
 - Customize data displayed or user authority with user exits.

IBM Health Checker for z/OS – Set up – 6. Create HZSPRMxx parmlib member

Create HZSPRMxx from the HZSPRM00 parmlib member

- At first, don't modify your HZSPRMxx parmlib member. Check to see what you get as output from IBM Health Checker for z/OS. Then, make permanent updates for your environment.
- HZSPRMxx should ONLY include:
 - Policy statements (for changes applied to added or refreshed checks)
 - LOGGER parameter (for indicating what log stream to use)
 - ...and when you want to add your own check, ADD or ADDREPLACE CHECK
- Other non-policy statements are ineffective, because the HZSPRMxx specified in your HZSPROC is processed before any checks are begun.
- HZSPRM00 is supplied in your DDDEF'd PARMLIB data set (and has no values used):

Remember, must start with HZS

```

/*  LOGGER=ON, LOGSTREAMNAME=HZSlogstream                                */
/*                                                                                   */
/*  {ADD | ADDREPLACE}, POLICY, STATEMENT=statementname, UPDATE, filters, */
/*  update_options, REASON=(reason text), DATE=yyyymmdd                       */

```


IBM Health Checker for z/OS – Set up – 7. Start IBM Health Checker for z/OS

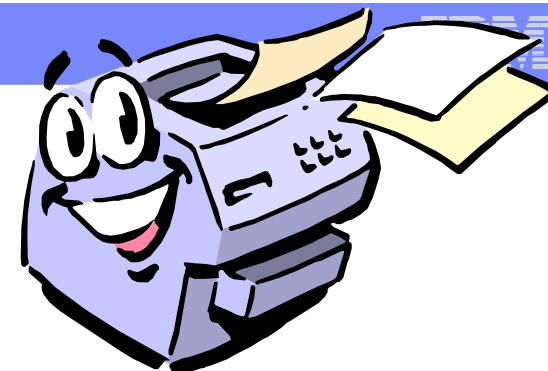
Start IBM Health Checker for z/OS

1. Copy SAMPLIB's HZSPROC into your proclib data set.
2. Update to point to your HZSPDATA data set and HZSPRMxx parmlib member you previously set up. (00 is good to start with.)
3. Update your COMMNDxx parmlib member with *COM='START HZSPROC'* (when ready to start it for each IPL) - **OR** -
4. Start the started task: *S HZSPROC*

```
//HZSPROC JOB JESLOG=SUPPRESS
//HZSPROC  PROC HZSPRM='00'
//HZSSTEP  EXEC   PGM=HZSINIT,REGION=0K,TIME=NOLIMIT,
//          PARM='SET PARMLIB=&HZSPRM'
//HZSPDATA DD   DSN=SYS1.&SYSNAME..HZSPDATA,DISP=OLD
//          PEND
//          EXEC HZSPROC
```

Your parmlib member you set up

Your persistent data set you set up



IBM Health Checker for z/OS – Set up – 8. Set up HZSPRINT utility

Set up the HZSPRINT utility to view print output

- HZSPRINT utility allows you to see check output in the message buffer.
 - May use SDSF instead (and SDSF allows you to see logstream history)
- HZSPRINT writes the selected message buffer for the target checks to SYSOUT.
- Use the HZSPRINT sample JCL from SYS1.SAMPLIB
 - If LOGSTREAM is specified, then RACF auth to the requested logstream is required.
 - If LOGSTREAM is not specified, then RACF auth to the requested check(s) is req.

```
//HZSPRINT EXEC PGM=HZSPRNT,TIME=1440,REGION=0M,
//      PARM=( 'CHECK(check_owner,check_name)' )
// *    PARM=( 'LOGSTREAM(logstreamname)' )
// *    'SYSNAME(sysname)',
// *    'CHECK(owner,name)' )
//SYSOUT DD SYSOUT=A,DCB=(LRECL=256)
```

Defaults to *,* (points to check_name)

Defaults to * (points to SYSNAME(sysname))

Only show exceptions (points to LOGSTREAM(logstreamname))

When using a data set, it must be seq, FB, LRECL 256 (points to DCB=(LRECL=256))

IBM Health Checker for z/OS – Set up – 9. Obtain checks

**Periodically, obtain checks for IBM Health Checker for z/OS
introduced via PTFs:**

1. SMP/E RECEIVE the most recent Enhanced HOLDDATA, which contains the latest FIXCAT information. (RECEIVE ORDER, ftp download site using FULL, ServiceLink,...)
2. Determine what checks are missing from your system:

```
SET BDY(GLOBAL).          /* Your z/OS global */  
REPORT MISSINGFIX ZONES(TGTZOS)  
FIXCAT(IBM.Function.HealthChecker).
```



Sample check output - SUCCESS

```
CHECK( IBMRSM,RSM_MAXCADS )
```

```
START TIME: 08/13/2010 14:33:26.251651
```

```
CHECK DATE: 20041006 CHECK SEVERITY: MEDIUM
```

```
CHECK PARM: THRESHOLD(80%)
```

```
IARH108I The current number of in use CADS entries is 25, which  
represents 62% of the total allowed CADS entries of 40. The  
highest
```

```
usage of CADS entries during this IPL is 65%, or 26 total  
entries. This
```

```
is below the current owner supplied threshold of 80%.
```

```
END TIME: 08/13/2010 14:33:26.252156 STATUS: SUCCESSFUL
```

Sample check output – Not Applicable

```
CHECK(IBMCS,CSVAM_T1BUF_T2BUF_NOEE)
```

```
START TIME: 08/11/2010 21:18:31.761670
```

```
CHECK DATE: 20060701 CHECK SEVERITY: LOW
```

```
ISTH019I This check is not applicable in the current VTAM  
environment.
```

```
Enterprise Extender (EE) lines have been activated on this system or  
VTAM Start Options associated with EE have been specified.
```

```
HZS1003E CHECK(IBMCS,CSVAM_T1BUF_T2BUF_NOEE):
```

```
THE CHECK IS NOT APPLICABLE IN THE CURRENT SYSTEM ENVIRONMENT.
```

```
END TIME: 08/11/2010 21:18:31.762012 STATUS: ENV N/A
```

Sample check output – EXCEPTION

```
CHECK(IBMASM,ASM_PLPA_COMMON_SIZE)
START TIME: 08/13/2010 15:21:43.391644
CHECK DATE: 20041006 CHECK SEVERITY: MEDIUM
CHECK PARAM: THRESHOLD(100%)
```

* Medium Severity Exception *

ILRH0105E PLPA/Common page data set size is below recommended value

Explanation: The size of CSA and ECSA is approximately 35389440 bytes which corresponds to 8640 auxiliary storage slots. The size of PLPA and EPLPA is approximately 76984318 bytes which corresponds to 18794 auxiliary storage slots. The size of the Common page data set is 26639 slots, the size of the PLPA page data set is 179 slots. The combined size of the PLPA and Common page data sets is 98% of the slots required for all of CSA/ECSA and PLPA/EPLPA. This is below the check warning threshold of 100%.

...

Some sample check outputs – EXCEPTION ...continued

Source: Aux Storage Manager

Reference Documentation:

"Auxiliary Storage Management Initialization" in z/OS
MVS

Initialization and Tuning Guide

Automation: N/A

Check Reason: To ensure that the PLPA and Common page data
sets are

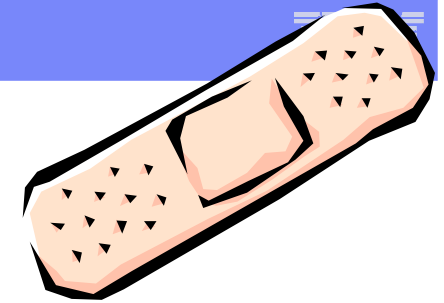
sized correctly

END TIME: 08/13/2010 15:21:43.411758 STATUS: EXCEPTION-MED

Some considerations about exceptions

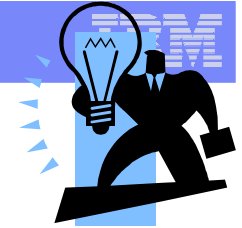
- You may have had your system configured a certain way for a long time, and now, IBM Health Checker for z/OS is telling you there are exceptions!
 - It's worth it to look over exceptions and make an evaluation. YOU need to decide what is appropriate for your system.
- Just because you get an exception, doesn't mean that there is a problem to report to IBM. Exceptions are meant to tell you about potential impacts.
- When you get an exception, it will be issued to:
 - Message buffer – you can view via SDSF or HZSPRINT
 - WTO message – by default. Format is HZS message number and qualified check name, followed on the next line by the check specific, 'actual' exception message id and text.
- **No automatic correction of exceptions is done by IBM Health Checker for z/OS.**





Resolving your exceptions

- Resolving your exceptions is important, or they will be repeatedly reported. (And probably get mixed up with ones you don't want to ignore!)
- You can resolve exceptions in the following ways:
 1. Make the suggested updates to your system. This will not result in an exception when the check runs again. This is the recommended way!
 2. Look at the values that the check is using. If the values aren't appropriate, update the values for your system. This will not result in an exception when the check runs again.
 3. Make the check *Inactive* or delete the check. (*Inactive* is a user-controlled state.)
- You can make dynamic, temporary changes (via SDSF or the MODIFY HZSPROC command) or permanent changes (HZSPRMxx).



Helpful suggestions for managing your checks

- If you don't want or need the messages appearing on your console, you could:
 - Set all checks' WTOTYPE to HARDCOPY.

```
UPDATE CHECK (* , * ) WTOTYPE (HARDCOPY)
```
- In addition, you might then have your automation package inform you based on different health checker messages. For instance:
 - HZS0002E to email the z/OS sysprogs (for medium severity exceptions)
 - HZS0003E to email and page the z/OS sysprogs and operators (for high severity exceptions)

This will inform you of just the multi-line WTO HZS messages, though, doesn't give you the entire check output...



An interesting idea SHARE'd by a resourceful user*

If you want to get the full text of the exceptions emailed to you, you could:

1. Have automation on the HZS multi-line WTO kick off a job
 - You know the name of the check that is getting the exception from the HZS message.
2. The job invokes HZSPRINT for that check, and also wraps some useful text before and after the HZSPRINT output, then queues it to SMTP mail.
 - For instance, you can put the check name in the Subject line.
3. SMTP mail sends you an email when you have an exception!

* *Sam Knutson, Geico Company, SHARE Orlando 2011, www.share.org, Session 9677 "Bit Bucket x"29", p.32-45*



Permanent changes

Simple example from the prior exception...making a permanent change to the parameter to resolve the problem:

- Modify or create HZSPRMyy:

```
ADDREPLACE POLICY(FIXEXEPT1)
      UPDATE
      CHECK(IBMASM,ASM_PLPA_COMMON_SIZE)
      PARM('THRESHOLD(90%)')
      REASON=('90% is good for us')
      DATE=(20100813)
ACTIVATE POLICY(FIXEXEPT1)
```

- F HZSPROC,REPLACE,PARMLIB=yy
 - To add this policy (with the check changes), and activate it.
- Then, update your HZSPROC if you used a new parmlib member:

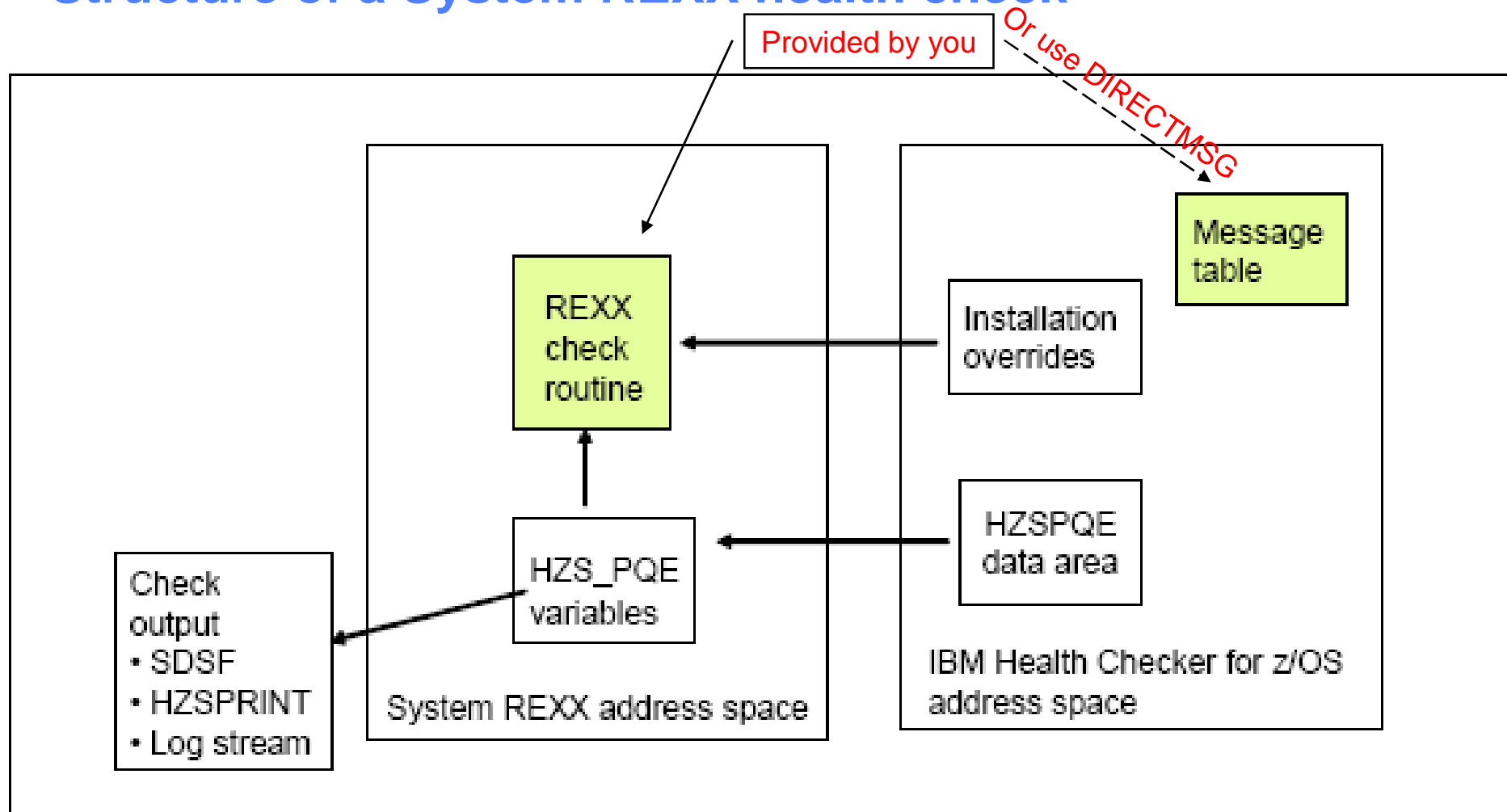
```
//HZSPROC JOB JESLOG=SUPPRESS
//HZSPROC  PROC HZSPRM='YY'
//HZSSTEP EXEC  PGM=HZSINIT,REGION=0K,TIME=NOLIMIT,
//          PARM='SET PARMLIB=&HZSPRM'
//HZSPDATA DD  DSN=SYS1.&SYSNAME..HZSPDATA,DISP=OLD
//          PEND
//          EXEC HZSPROC
```

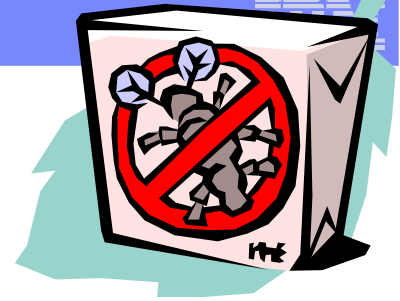


Writing your own checks

1. **Decide: local, remote, System REXX? Assembler or Metal C?**
 - Samples can be found in SYS1.SAMPLIB (and /usr/lpp/bcp/samples for Metal C)!
2. **Write your check routine to do what you want it to do. Check names are important, for handling them as a group with wildcarding.**
3. **Create a message table for the check output. As of z/OS R12 this is easier with the DIRECTMSG enhancement, which needs no message table.**
4. **Ensure that the check routine (and message table, if used) is available to the system - in LNKLST, or in a System REXX concatenated data set for System REXX checks.**
5. **Add the check to IBM Health Checker for z/OS:**
 - a) Create a HZSADDCHECK exit routine. This routine adds one or more checks and provides default values. It is called by IBM Health Checker for z/OS dynamic exit, HZSADDCHECK, then
 - b) Add HZSADDCHECK exit routine to the HZSADDCHECK exit, and have system run the exit routine. For testing, can do it via operator command (SETPROG EXIT,ADD ...).
 - c) Then, have Health Checker pick up the exit routine right away: F HZSPROC,ADDNEW or ?HZSCHECK REQUEST(ADDNEW).
 - **Or for local and System REXX checks only (which can avoid exit routines):**
 - In HZSPRMxx: ADDREPLACE CHECK(...)
6. **Check will then run on its own, at the interval you specified for the check.**

IBM Health Checker for z/OS – Structure of a System REXX health check





System REXX checks and DEBUG

- You can turn the DEBUG parameter on for a check via SDSF, F HZSPROC,UPDATE,CHECK=(..,..),DEBUG=ON, or HZSPRMxx
- System REXX check important keywords in HZSPRMxx and HZSADDCK are:
 - REXXTSO(YES|NO): whether to run in a TSO environment
 - REXXIN(YES|NO): for non-TSO environment only, if an input data set is to be used
 - REXXTIMELIMIT(nn): how many seconds it can run before the system will end it. 0 is the default; means no limit.
 - REXXHLQ(hlq_name): hlq to be used for any input or output for the check
 - REXXIN – rexxhlq.execname.REXXIN[.E#####]
 - REXXOUT – rexxhlq.execname.REXXOUT[.E#####]
 - Where E##### used if ENTRYCODE(>0) specified when check was defined
- **REXXOUT DSN allocated when check is run in DEBUG mode.** You can use SAY and TRACE to write to this data set.



Advanced Topics: New Items in z/OS R12 and R13!

Health Checker Framework improvements

- R12: METAL C headers for Health Checker and sample check
- R12: Health check history display in SDSF
- R12: Simpler message interface (DIRECTMSG)
- R12: Run with assigned userid that has access to BPX.SUPERUSER (that is, not required to be UID(0))
- R13: New SYNCVAL keyword in HZSPRMxx parmlib member and MODIFY, to indicate when to run the check (SYSTEM , hh:mm, *:mm). Can be used with existing INTERVAL and EINTERVAL for better check scheduling.
- R13: Checks can raise message severity as conditions change (for instance, thresholds are getting closer)
- R13: DOM control for checks, for suppressing WTOs and any automation actions they trigger for a check that is generating exceptions
- ...and of course, always new health checks !!!

Introduction to IBM Health Checker for z/OS Summary

- **IBM Health Checker for z/OS, it's in your z/OS release!**
- **Setting up IBM Health Checker for z/OS**
 1. Satisfy software requirements for IBM Health Checker for z/OS, as needed
 2. **Allocate the HZSPDATA data set to save check data between restarts**
 3. Define log streams to keep a record of the check output, as needed
 4. Create security definitions
 5. **Helpful: Set up customization and security for SDSF support for IBM Health Checker for z/OS in IBM Health Checker in z/OS SDSF Operation and Customization**
 6. Create HZSPRMxx from the HZSPRM00 parmlib member
 7. **Start IBM Health Checker for z/OS**
 8. Set up the HZSPRINT utility, as needed
 9. Obtain checks for IBM Health Checker for z/OS
- **Know what check exceptions are, and what to do about them**
- **Write your own checks!**





BACK UP SLIDES:

**Screen shots for setting up IBM Health Checker
for z/OS**

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

File Edit Edit_Settings Menu Utilities Comp



```

EDIT          SYS1.SAMPLIB(HZSALLCP) - 01.00
Command ==>
***** ***** Top of Data *****
000110 //HZSALLCP JOB 'D98A,B9211068','UTILITY JOB',
000120 //          MSGLEVEL=(1,1),CLASS=A,MSGCLASS=H
000130 //*
000200 //*****
003600 //*****
003700 //*
003800 //HZSALLCP EXEC PGM=HZSAIEOF,REGION=4096K,TIME=1440
003900 //HZSPDATA DD DSN=SYS1.SY1.HZSPDATA,DISP=(NEW,CATLG),
004000 //          SPACE=(4096,(100,400)),UNIT=SYSALLDA,VOL=SER=PAGE08,
004100 //          DCB=(DSORG=PS,RECFM=FB,LRECL=4096)
004200 //SYSPRINT DD DUMMY
***** ***** Bottom of Data *****

```

MA d 04/015

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

RACF Setup for HZSPROC

```

File Edit Edit_Settings Menu Utilities Compilers
EDIT      $$DRIVER.INFO(HC) - 01.01      Columns 00001 00072
Command ==>                               Scroll ==> PAGE
***** ***** Top of Data *****
000001 //HCRACF  JOB 'D98A,B9211068','UTILITY JOB',
000002 //                MSGLEVEL=(1,1),CLASS=A,MSGCLASS=H
000003 //RACFTGT1 EXEC PGM=IKJEFT01
000004 //SYSTSPRT DD  SYSOUT=*
000005 //SYSTEM   DD  DUMMY
000006 //SYSTSIN  DD  *
000007         ADDUSER      +
000008             HZSPROC    +
000009             OMVS(UID(0) HOME('/ /') PROGRAM('/bin/sh'))  +
000010             NOPASSWORD  +
000011             NOOIDCARD  +
000012             DFLTGRP(SYS1)
000013             CONNECT HZSPROC +
000014             GROUP(SYS1) +
000015             AUTH(USE)
000016             RDEFINE           +
000017             STARTED           +
000018             HZSPROC.*         +
    
```

MA d

09/002

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

```

Session D - [24 x 80]
File Edit View Communication Actions Window Help
File Edit Edit_Settings Menu Utilities Comp

VIEW      SYS1.PARMLIB.INSTALL(HZSPRM00) - 01.00
Command ==>
Scroll ==> PAGE
***** Top of Data *****
000001 /* ----- */
- - - - - 16 Line(s) not Displayed
000018 /*  LOGGER=ON,LOGSTREAMNAME=HZSlogstream */
- - - - - 5 Line(s) not Displayed
000024 /*  {ADD | ADDREPLACE},POLICY,STATEMENT=statementname,UPDATE,filters,*/
000025 /*      update_options,REASON=(reason text),DATE=yyyymmdd */
- - - - - 63 Line(s) not Displayed
000089 /* ----- */
***** Bottom of Data *****

MA d 04/015
Connected to remote server/host s390vm.pok.ibm.com using port 23
    
```

Setting up HZSPRM00

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

File Edit Edit_Settings Menu Utilities Compile

VIEW SYS1.PROCLIB.POK(HZSPROC) - 01.00 Columns 00001 00072
Command ==> Scroll ==> PAGE

```
***** ***** Top of Data *****  
000001 //HZSPROC JOB JESLOG=SUPPRESS  
000002 //*****/  
- - - - - 32 Line(s) not Displayed  
000035 //*****/  
000036 //HZSPROC PROC HZSPRM='00'  
000037 //HZSSTEP EXEC PGM=HZSINIT,REGION=0K,TIME=NOLIMIT,  
000038 // PARM='SET PARMLIB=&HZSPRM'  
000039 //HZSPDATA DD DSN=SYS1.&SYSNAME..HZSPDATA,DISP=OLD  
000040 // PEND  
000041 // EXEC HZSPROC  
***** ***** Bottom of Data *****
```

Setting up
HZSPROC

MA

d

09/002

DEMO – Setting up and using IBM Health Checker for z/OS

Session C - [32 x 80]

File Edit View Communication Actions Window Help

```

RELEASE z/OS 01.07.00 LICENSE = z/OS
USED LOADMS IN SYS1.PARMLIB ON 0980
ARCHLVL = 2 MTLSHARE = N
IEASYM LIST = (MS,L)
IEASYS LIST = (MS) (OP)
IODF DEVICE 02FF
IPL DEVICE 0980 VOLUME ZDR17B
- SY1 d u,,,980,8
SY1 IEE457I 17.00.11 UNIT STATUS 777
UNIT TYPE STATUS VOLSER VOLSTATE
0980 3390 S ZDR17B PRIV/RSDNT
0981 3390 A PAGE08 STRG/RSDNT
0982 3390 A SPOOL1 STRG/RSDNT
0983 3390 D CICSPK PRIV/RSDNT
0984 3390 F-NRD /RSDNT
0985 3390 A ALPTCP PRIV/RSDNT
0986 3390 D D97107 PRIV/RSDNT
0987 3390 F-NRD /RSDNT
00- SY1 d a,l
SY1 IEE114I 17.00.17 2005.218 ACTIVITY 780
JOBS M/S TS USERS SYSAS INITS ACTIVE/MAX VTAM DAS
00002 00007 00001 00026 00004 00001/00300 00004
LLA LLA LLA NSW S VLF VLF VLF NSW S
VTAM VTAM VTAM NSW S TCPIP TCPIP TCPIP NSW SO
JES2 JES2 IEFPROC NSW S RACF RACF RACF NSW S
TCAS TCAS TSO OWT S INETD1 STEP1 INETD OWT AO
FTPD1 STEP1 FTPD OWT AO
IBMUSER OWT
IEE612I CN=C3E0SY1 DEVNUM=03E0 SYS=SY1
-
IEE163I MODE= RD
    
```



MA c

30/003

Connected to remote server/host s390vm.pak.ibm.com using port 23

DEMO – Setting up and using IBM Health

It's there now!

With an
exception ☹️

Session C - [32 x 80]

File Edit View Communication Actions Window Help

```

- SY1 HZS0010I THE HZSPDATA DATA SET CONTAINS NO RECORDS
SY1 HZS0403I SET PARMLIB PROCESSING HAS BEEN COMPLETED
- SY1 HZS0103I HZSPROC INITIALIZATION COMPLETE
SY1 HZS0001I CHECK(IBMUSS,USS_MAXSOCKETS_MAXFILEPROC):
BPXH032E MaxFileProc value is too low.
SY1 HZS0001I CHECK(IBMUSS,USS_MAXSOCKETS_MAXFILEPROC):
BPXH033E MaxSockets value for AF_INET is too low.
SY1 HZS0001I CHECK(IBM CNZ,CNZ_AMRF_EVENTUAL_ACTION_MSGS):
CNZHF0004I Retaining eventual action messages may consume storage
needed by critical or immediate action messages.
SY1 HZS0002E CHECK(IBM XCF,XCF_TCLASS_CLASSLEN):
IXCH0420E Transport class definitions do not provide sufficient size
segregation
SY1 HZS0001I CHECK(IBM CNZ,CNZ_CONSOLE_ROUTCODE_11):
CNZHF0005I One or more consoles are configured to receive messages
intended only for programmers.
SY1 HZS0001I CHECK(IBM RSM,RSM_MEMLIMIT):
IARH109E MEMLIMIT SET TO MAXIMUM
SY1 HZS0002E CHECK(IBM SDUMP,SDUMP_AUTO_ALLOCATION):
IEAH701I SDUMP is not using automatic allocation.
SY1 HZS0001I CHECK(IBM RSM,RSM_REAL):
IARH101E V=R Storage is Defined
SY1 HZS0001I CHECK(IBM CNZ,CNZ_SYSCONS_ROUTCODE):
CNZHF0008I System console SY1 is not configured to receive the
minimum set of routing codes (1, 2, and 10).
00 *SY1 *HZS0003E CHECK(IBM RACF,RACF_SENSITIVE_RESOURCES):
*IRRH204E The RACF_SENSITIVE_RESOURCES check has found one or more
*potential errors in the security controls on this system.
IEE612I CN=C3E0SY1 DEVNUM=03E0 SYS=SY1

IEE163I MODE= RD

```

MA c

30/003

Connected to remote server/host s390vm.pak.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

```
Display Filter View Print Options Help
-----
HQX7720 ----- SDSF PRIMARY OPTION MENU -----
COMMAND INPUT ==> ck                                SCROLL ==> PAGE

DA   Active users          INIT  Initiators
I    Input queue          PR    Printers
O    Output queue         PUN   Punches
H    Held output queue    RDR   Readers
ST   Status of jobs      LINE  Lines
                                     NODE  Nodes
LOG  System log          SO    Spool offload
SR   System requests     SP    Spool volumes
MAS  Members in the MAS
JC   Job classes         RM    Resource monitor
SE   Scheduling environments
RES  WLM resources      CK    Health checker
ENC  Enclaves           ULOG  User session log
PS   Processes

END   Exit SDSF
```

Looking at the results

MA

d

05/023

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

```

Display Filter View Print Options Help
-----
SDSF HEALTH CHECKER DISPLAY SY1                               LINE 1-18 (48)
COMMAND INPUT ==>                                           SCROLL ==> HALF
NP  NAME                                                    e      Status          Result D
   CNZ_AMRF_EVENTUAL_ACTION_MSGS      VE(ENABLED)    EXCEPTION-LOW    4 0
   CNZ_CONSOLE_MASTERAUTH_CMDSYS      VE(ENABLED)    SUCCESSFUL       0 0
   CNZ_CONSOLE_MSCOPE_AND_ROUTCODE    VE(DISABLED)   ENV N/A         0 0
   CNZ_CONSOLE_ROUTCODE_11            VE(ENABLED)    EXCEPTION-LOW    4 0
   CNZ_EMCS_HARDCOPY_MSCOPE           VE(DISABLED)   ENV N/A         0 0
   CNZ_EMCS_INACTIVE_CONSOLES         VE(ENABLED)    SUCCESSFUL       0 0
   CNZ_SYSCONS_MASTER                  VE(ENABLED)    SUCCESSFUL       0 0
   CNZ_SYSCONS_MSCOPE                  VE(DISABLED)   ENV N/A         0 0
   CNZ_SYSCONS_PD_MODE                 VE(ENABLED)    SUCCESSFUL       0 0
   CNZ_SYSCONS_ROUTCODE                VE(ENABLED)    EXCEPTION-LOW    4 0
   CNZ_TASK_TABLE                      VE(ENABLED)    SUCCESSFUL       0 0
   GRS_CONVERT_RESERVES                VE(DISABLED)   ENV N/A         0 0
   GRS_EXIT_PERFORMANCE                VE(ENABLED)    SUCCESSFUL       0 0
   GRS_MODE                             VE(DISABLED)   ENV N/A         0 0
   GRS_SYNCHRES                        VE(ENABLED)    SUCCESSFUL       0 0
   RACF_GRS_RNL                        VE(DISABLED)   ENV N/A         0 0
   RACF_SENSITIVE_RESOURCES            VE(ENABLED)    EXCEPTION-HIGH   12 0
   RSM_AFQ                              VE(ENABLED)    SUCCESSFUL       0 0

```

There are some exceptions

MA

d

05/021

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

Playing around to learn

```

Display Filter View Print Options Help
-----
SDSF HEALTH CHECKER DISPLAY SY1          COMMAND ISSUED
COMMAND INPUT ==>                        SCROLL ==> HALF
NP  NAME                                     il Severity SevCode  WType ModifiedBy
CNZ_AMRF_EVENTUAL_ACTION_MSGS             0 LOW 4 INFO
CNZ_CONSOLE_MASTERAUTH_CMDSYS             0 LOW 4 INFO
CNZ_CONSOLE_MSCOPE_AND_ROUTCODE           0 LOW 4 INFO
CNZ_CONSOLE_ROUTCODE_11                   0 LOW 4 INFO
CNZ_EMCS_HARDCOPY_MSCOPE                   0 MEDIUM 8 EVENTUAL
CNZ_EMCS_INACTIVE_CONSOLES                 0 HIGH 12 CRITICAL
CNZ_SYSCONS_MASTER                         0 HIGH 12 CRITICAL
CNZ_SYSCONS_MSCOPE                         0 MEDIUM 8 EVENTUAL
CNZ_SYSCONS_PD_MODE                        0 MEDIUM 8 EVENTUAL
CNZ_SYSCONS_ROUTCODE                       0 LOW 4 INFO
CNZ_TASK_TABLE                             0 LOW 4 INFO
GRS_CONVERT_RESERVES                       0 LOW 4 INFO
GRS_EXIT_PERFORMANCE                       0 LOW 4 INFO
GRS_MODE                                    0 LOW 4 INFO
GRS_SYNCHRES                               0 LOW 4 INFO
RACF_GRS_RNL                               0 HIGH 12 CRITICAL
RACF_SENSITIVE_RESOURCES                   0 LOW 4 INFO MODIFY COMMA
RSM_AFQ                                    0 HIGH 12 CRITICAL
    
```

MA

d

23/002

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

```

Display Filter View Print Options Help
-----
SDSF HEALTH CHECKER DISPLAY SY1
COMMAND INPUT ==>
NP NAME tart-Date-Time Interval NextSch-Date-T
CNZ_AMRF_EVENTUAL_ACTION_MSGS 8/06/2005 17:02:14 24:00 08/07/2005 17:
CNZ_CONSOLE_MASTERAUTH_CMDSYS 8/06/2005 17:02:14 24:00 08/07/2005 17:
CNZ_CONSOLE_MSCOPE_AND_ROUTCODE 8/06/2005 17:02:14 24:00 ***** N/A ***
CNZ_CONSOLE_ROUTCODE_11 8/06/2005 17:02:14 24:00 08/07/2005 17:
CNZ_EMCS_HARDCOPY_MSCOPE 8/06/2005 17:02:14 24:00 ***** N/A ***
CNZ_EMCS_INACTIVE_CONSOLES 8/06/2005 17:02:14 24:00 08/07/2005 17:
CNZ_SYSCONS_MASTER 8/06/2005 17:02:14 24:00 08/07/2005 17:
CNZ_SYSCONS_MSCOPE 8/06/2005 17:02:14 24:00 ***** N/A ***
CNZ_SYSCONS_PD_MODE 8/06/2005 17:02:14 1:00 08/06/2005 18:
CNZ_SYSCONS_ROUTCODE 8/06/2005 17:02:14 24:00 08/07/2005 17:
CNZ_TASK_TABLE 8/06/2005 17:17:14 0:15 08/06/2005 17:
GRS_CONVERT_RESERVES 8/06/2005 17:02:14 ONETIME ***** N/A ***
GRS_EXIT_PERFORMANCE 8/06/2005 17:02:14 24:00 08/07/2005 17:
GRS_MODE 8/06/2005 17:02:14 ONETIME ***** N/A ***
GRS_SYNCHRES 8/06/2005 17:02:14 1:00 08/06/2005 18:
RACF_GRS_RNL 8/06/2005 17:02:14 8:00 ***** N/A ***
RACF_SENSITIVE_RESOURCES 8/06/2005 17:19:27 0:01 08/06/2005 17:
RSM_AFQ 8/06/2005 17:02:14 ONETIME ***** N/A ***

```

Changing the interval

MA

d

23/002

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

```

Display Filter View Print Options Help
-----
SDSF SYSLOG      2.101 SY1  SY1  08/06/2005 0W      2134
COMMAND INPUT ==>
17:18:12.20 IBMUSER 00000290 F HZSPROC,UPDATE,CHECK=(IBMRACF,RACF_SENSITIVE_R
WTO=INFO
17:18:12.21 STC00022 00000090 HZS0400I CHECK(IBMRACF,RACF_SENSITIVE_RESOURCES)
809 00000090 UPDATE PROCESSING HAS BEEN COMPLETED
17:19:28.60 IBMUSER 00000290 F HZSPROC,UPDATE,CHECK=(IBMRACF,RACF_SENSITIVE_R
17:19:28.60 STC00022 00000090 HZS0400I CHECK(IBMRACF,RACF_SENSITIVE_RESOURCES)
811 00000090 UPDATE PROCESSING HAS BEEN COMPLETED
17:19:33.14 STC00022 00000090 HZS0001I CHECK(IBMRACF,RACF_SENSITIVE_RESOURCES)
812 00000090 IRRH204E The RACF_SENSITIVE_RESOURCES check has
812 00000090 potential errors in the security controls on thi
17:20:37.66 STC00022 00000090 HZS0001I CHECK(IBMRACF,RACF_SENSITIVE_RESOURCES)
813 00000090 IRRH204E The RACF_SENSITIVE_RESOURCES check has
813 00000090 potential errors in the security controls on thi
17:21:42.19 STC00022 00000090 HZS0001I CHECK(IBMRACF,RACF_SENSITIVE_RESOURCES)
814 00000090 IRRH204E The RACF_SENSITIVE_RESOURCES check has
814 00000090 potential errors in the security controls on thi
17:22:46.71 STC00022 00000090 HZS0001I CHECK(IBMRACF,RACF_SENSITIVE_RESOURCES)
815 00000090 IRRH204E The RACF_SENSITIVE_RESOURCES check has
815 00000090 potential errors in the security controls on thi
. . . . .
MA d 04/021
    
```

Just like I wanted...

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

So much power
in SDSF

```
HELP: Health Checker Panel
COMMAND INPUT ==>

Select a topic by number, or press Enter to view them in sequence.

  1 - Introduction to the CK panel
  2 - Syntax of the CK command
  3 - Action characters: browse, pause, delete, etc.
  4 - Fields on the CK panel
  5 - Overtyping fields to change their values
  6 - Commands: sort, filter, search, etc.

These topics are displayed only if selected:

  97 - What's new
  98 - Display the table of contents, index, or tutorial
  99 - Messages

F1 = Help           F10 = Previous       Enter = Forward
F3 = Exit           F7 = Up              TOC = Menu

MA d 04/023
```

Connected to remote server/host s390vm.pok.ibm.com using port 23

DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

File Edit Edit_Settings Menu Utilities Compilers

EDIT SYS1.SAMPLIB(X) - 01.00

Columns 00001 00072

Command ==> Scroll ==> PAGE

```
***** ***** Top of Data *****
000100 //HCPRI NT JOB 'D98A,B9211068','UTILITY JOB',
000200 //          MSGLEVEL=(1,1),CLASS=A,MSGCLASS=H
000300 // *
000400 //*****/
000500 //HZSPRI NT EXEC PGM=HZSPRI NT,TIME=1440,REGION=0M,
000600 //          PARM=('CHECK(*,*)')
000700 //SYSOU T DD SYSOU T=A,DCB=(LRECL=256)
***** ***** Bottom of Data *****
```

Using
HZSPRI NT

MA d

05/015

Connected to remote server/host s390vm.pok.ibm.com using port 23

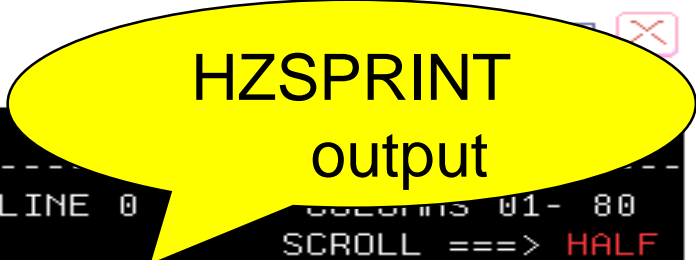
DEMO – Setting up and using IBM Health Checker for z/OS

Session D - [24 x 80]

File Edit View Communication Actions Window Help

Display Filter View Print Options Help

```
-----
SDSF OUTPUT DISPLAY HCPRIINT  JOB00023  DSID   101 LINE 0  SCROLLS 01- 80
COMMAND INPUT ==>                                SCROLL ==> HALF
***** TOP OF DATA *****
```



```
1*****
*
* HZSPRINT (UA19064-05175)  2005/08/06 17:33
*
* HZSU001I Check messages
* Sysplex: LOCAL      System: SY1
*
* Filter: CHECK(*,*)
*
*****
```

```
*****
*
* Start: CHECK(IBMGRS,GRS_SYNCHRES)
*
*
* . . . . .
```

MA d 04/021

Connected to remote server/host s390vm.pok.ibm.com using port 23

Introduction and Getting Started with the IBM Health Checker for z/OS

