

Xymon Client for z/VSE Installation and Configuration Guide

**Version 1.2
February, 2009**

Table of Contents

Copyright.....	3
Changes.....	3
Introduction.....	4
Design Criteria.....	4
Changes Since V1.1.....	4
Installation of the Client.....	5
Contents of the Package.....	6
Configuring the Client.....	7
Operating the Client.....	13
Acknowledgments.....	13

Copyright

Xymon Client for z/VSE
Copyright (C) 2003-2009 Richard Smrcina

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

<http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>

Changes

2008/10/31 – Redesigned documentation for the release of client V1.2
2009/02/11 – Xymon name change

Introduction

Xymon is an open source network services monitor. It runs on Linux (including Linux for System z). The Xymon Client for z/VSE will perform tests on a z/VSE system and report the status of those tests back to the Xymon server.

The client is written mostly in REXX with a number of assembler based function and a COBOL program. The client uses TCP/IP sockets to send the status message(s) to the Xymon server.

Xymon is written and Copyright © 2007-2009 Henrik Storner.

Design Criteria

The Xymon Client for z/VSE is not intended as a performance monitor. The fact that it gathers some performance metric data is rooted in the nature of Xymon's role as a network host threshold monitor. It collects information every five minutes and reports it back to the Xymon server; this is much too sparse for true performance monitoring. It should not take on the task of a performance monitor as that is not its purpose.

Changes Since V1.1

Note: If you are currently running version 1.1 of the client, a configuration change is required.

The DISK parameters in HOBVARS now require the fileid of the VSAM catalog.

If you have the following coded:

```
DISK CAT=VSESP.USER.CATALOG
```

it needs to be changed to this:

```
DISK CAT=VSESPUC/VSESP.USER.CATALOG
```

The disk results web page and graphs will contain the fileid, while the VSAM catalog is checked with the data set name.

By far the biggest change is that the client can now take advantage of Xymon's client data stream to report the test results. This allows configuration of threshold values at the Xymon server (centralized configuration) instead of at the client (local

configuration). This configuration parameter is set in the HOBVARS file (see below).

The Xymon client for z/VSE can take advantage of the Xymon client data stream. This allows configuration of most of the thresholds to be done on the Xymon server, also called centralized configuration. This mode of operation requires Xymon Server V4.3 to be running on the machine hosting Xymon. This is an optional change, if it is preferred to set thresholds on VSE, that functionality is still available.

The DISK test has been changed to represent the results in Linux 'df' format. This change was first implemented in the z/VM client and suggested by Thomas Kern, a consultant for the DOE.

The VSESPool test has been merged into the DISK test.

A CICS test has been added which will show and graph the CICS transaction rate and will graph the DSA Utilization on the Xymon trends page.

Interface traffic monitoring and ports monitoring has been implemented for the TCP/IP Tools stack by Barnard Software.

A memory test has been added to show the virtual storage utilization. A test has been added to show the number of running partitions against the NPARTS IPL configuration.

Multiple bbdDisplays are now supported.

Installation of the Client

The entire package is distributed as a zip file. Within the zip file are the programs that make up the client. The files with the .REXX extension will need to be put into a VSE library as .PROC members (either via FTP or your favorite VM or VSE development environment). The files with the .ASSEMBLE extension will need to be assembled, with the resulting phase put into a sublib in the LIBDEF PHASE search chain that the job running the client can access. Files with the .SAMP extension are parameter members and should be cataloged as .Z members into a sublib for configuration data (like PRD2.CONFIG).

Unzip the distributed archive. On a Linux system use 'unzip -a' to convert the CR/LF's properly.

Upload all files to your favorite development environment (ICCF, CMS, Bim-Edit, etc). Catalog all .REXX members to a VSE sublibrary as .PROC members.

The VSE client code is a long running job. A partition should be set aside for its use. The partition does not need to be large (6MB is sufficient).

The VSE client code uses IBM REXX Sockets, which needs to be enabled. In order to do this the REXX parameters phase will need to be modified. A sample of the default REXX parameters source program is in PRD1.BASE as ARXPARMS.Z. The comments in the code show how to enable REXX sockets. When the program is assembled it is put into PRD2.CONFIG by default. When the client is run, make sure that PRD2.CONFIG is at the beginning of the LIBDEF PHASE search chain.

The .SAMP members can be renamed or copied as .Z members. Make any configuration changes to the .Z members that are required for your site. Catalog all .Z members to the sublibrary that you specify in the PARM parameter of the EXEC statement for the driver. If no parm value is provided, the default is PRD2.CONFIG.

Use a standard batch program assembly on all .ASSEMBLE members. HOBVSCPU requires the generation feature to be installed and PRD2.GEN1 must be in the LIBDEF source search string for the assembly. Use a standard CICS COBOL compile for HOBICOL. Define the program to CICS, define the file HOBICOL to CICS and point the DLBL to the data set in the define based on site standards (if the define was changed from what is distributed). Add the program HOBICOL to the CICS startup PLT for any CICS that is to be monitored.

Set up HOBDSKTS on your job scheduler (if you have one) or use the distributed POWER Event Scheduling parameters to run the job. Adjust the frequency as you see fit. Submit the job.

If your system uses a TCP/IP stack identifier other than 00, you will need to modify the //OPTION SYSPARM in the Xymon driver job.

Submit the Xymon driver job, XYMONVSE. It will need a long running partition. It should also be released as part of the normal VSE startup process.

Installation is complete. Configuration can be started.

Contents of the Package

Figure 1 shows and describes the purpose of each member that is distributed.

<i>Program Name</i>	<i>Description</i>
HOBVSECI.REXX	Reads the VSAM file written by the CICS portion of the client and reports the data to the Xymon server.
HOBVSEDR.REXX	Driver for the Xymon VSE client. It controls the execution of the tests.
HOBVSEDU.REXX	Performs the disk utilization test.
HOBVSEGV.REXX	Performs the partition getvis utilization test.

<i>Program Name</i>	<i>Description</i>
HOBVSEIF.REXX	Performs the IP Interface Utilization test. This is only available when using the TCP/IP stack available from Barnard Software.
HOBVSEME.REXX	Performs the memory test. Only available with z/VSE 4.1 +DY46857.
HOBVSEPG.REXX	Performs the paging test.
HOBVSEPR.REXX	Performs the procs (running jobs) test.
HOBDSKTS.REXX	Collects VSAM catalog, VSE library and VTOC utilization data for use by the disk utilization test.
HOBTEST.SAMP	The main list of tests that the client will perform and the name of the REXX program used to run that test. Whether a certain test is run is controlled here via commenting out one or more lines in this file.
HOBVARS.SAMP	Variables used during the execution of the client. These include the IP Address of the Xymon server and the various thresholds that determine the color used in the status message sent to the server. Also, the volumes, VSAM catalogs and libraries for the disk utilization test and the job names for the processes test are listed here.
HOBDSKTS.JCL	Job to run Xymon disk utilization test (HOBDSKTS). It uses POWER Job Event Scheduling to run the job every hour.
HOBVSBODY.ASSEMBLE	Assembler program used as a REXX function. It is called by HOBVSEME to get system utilization information. It requires z/VSE 4.1 +DY46857. DO NOT attempt to use it without the fix for that APAR or any lower release!
HOBVSGVS.ASSEMBLE	Assembler program used as a REXX function to retrieve partition information from the GETFLD macro. Used by HOBVSEGV.
HOBVSCPU.ASSEMBLE	Assembler program used as a REXX function to get the CPU information from the TDSERV macro. The CPU data is stored in a librarian member temporarily to be retrieved by the next cycle to determine utilization.
HOBVSPAG.ASSEMBLE	Assembler program used as a REXX function to get the paging information from the GETFLD macro. The paging data is stored in a librarian member temporarily to be retrieved by the next cycle to determine the page rate.
HOBVSSYS.ASSEMBLE	Assembler program used as a REXX function to get the NPARTS IPL parameter value to compare against the number of running jobs. Used only in HOBVSEDR.
RXBSTCMD.ASSEMBLE	Assembler program used as a REXX function to pass a TCP/IP tools command and return the results to REXX. It is the basis for the PORTS and interface utilization features.
HOBCICOL.CBL	COBOL program that runs in any CICS region that will write transaction rate and DSA Utilization data to a VSAM file to report back to the Xymon server.
DEFHOBIT.JCL	JCL to define the VSAM file used by the CICS portion of the client.
HOBVSE.JCL	JCL used to run the client

Figure 1 - List of files distributed with the package

Configuring the Client

Configuration of the Xymon Client for z/VSE involves modification of the HOBTEST and HOBVARS members.

HOBTEST is used to determine which tests to run. The first value on each line is the name of the test being performed. The second value is the name of the REXX program used to perform that test.

Comment out the tests that you don't wish to or can't use. For instance, if you can't run the memory test because of your VSE level, it will need to be commented out by putting a pound-sign (#) in column 1 of the line.

The HOBTEST file is shown in Figure 2.

GETVIS	HOBVSEGV	/* System and partition GETVIS test	*/
PROCS	HOBVSEPR	/* Process test (running jobs)	*/
DISK	HOBVSEDU	/* VSAM and Volume Utilization	*/
CICS	HOBVSECI	/* CICS Transaction Rate and DSA	*/
CPU	HOBVSECP	/* CPU Utilization	*/
IFACE	HOBVSEIF	/* IP Interface Data (BSI Stack only)	*/
MEMORY	HOBVSEME	/* Memory Test (z/VSE 4.1 +PTF)	*/
PAGING	HOBVSEPG	/* Paging	*/

Figure 2 – Distributed HOBTEST file

The HOBVARS file is used to set up variables for the client and to configure threshold values for the various tests. It is displayed in Figure 3.

```
# Xymon z/VSE client options member
#
# Address of Xymon Server
bbdisplay 192.168.201.12
# Xymon port number
bbport 1984
# Hostname (use when gethostname function doesn't work)
hostname vse2
# For the GETVIS test specify the partition IDs that
# should be checked for GETVIS storage utilization
GETVIS SVA F1 Z1 D1 01
# The threshold values below are used to determine which color
# is sent back to Xymon. A measured value below the first
# number will return green. A measured value greater than or equal
# to the first number below the second number will return yellow.
# A measured value greater than the second number will return red.
# Test Yellow Red
DISK DEF 90 95
DISK VOL=DOSRES
DISK VOL=SYSWK1
DISK CAT=VSESPUC/VSESP.USER.CATALOG
DISK LIB=IJSYSRS
DISK LIB=PRD1
DISK LIB=PRD2 97 98
# The remaining entries are only required if you are using client side
```



```

# threshold monitoring. If threshold monitoring is being controlled
# by the Xymon server these lines can be deleted.
CPU      80    95
PAGING   3     5
SPOOL    80    90

# The PROCS test will display how many jobs are currently running
# and verify that certain jobs are running. The first value is an
# optional threshold of how many jobs are running. If that value is
# exceeded, a yellow page is generated. The next item is a library
# member that contains the job names that must be running.

# If it is determined that a required job is not running a red page
# will be generated.
PROCS 15
PROCS VTAMSTRT
PROCS CICSICCF

```

Figure 3 – Sample HOBVARS file

The CPU, PAGING and SPOOL test parameters contain values called yellow and red. These are percentage threshold values. If the measured percentage is less than the yellow value, a green status is returned to Xymon. If the measured percentage is between the two values inclusive, a yellow status is returned to Xymon. If the measured percentage is equal to or higher than the second value, a red status is returned to Xymon.

The meaning of the variables and test parameters are described in Figure 4.

<i>Variable Name</i>	<i>Usage</i>
bbdisplay	The IP Address of the Xymon Server. Multiple bbdisplays are supported to send client data to more than one Xymon server. Separate the IP addresses of the bbdisplays with one or more spaces. The port number can also be specified for each bbdisplay by adding a colon and the port number to the IP Address, eg: 192.168.1.72:1984.
bbport	The port number used to communicate to the Xymon Server (the default is 1984). This serves as the default for each bbdisplay specified in the bbdisplay variable. It must be specified even if the port number is coded explicitly with the IP Address.
hostname	The hostname of the VSE machine. It is only required to specify this value if the gethostname call does not work properly.
CPU	The CPU Utilization threshold. The CPU Utilization is obtained by calling a provided REXX function which is an assembler program that uses the TDSERV macro. TDSERV is only available on systems with the Turbo Dispatcher available and the generation feature. If you do not have the generation feature installed, contact me and I will send you a pre-compiled version of the program. If server based configuration is being used, the CPU variable is not required and the CPU threshold is configured with the LOAD parameter in hobbit-clients.cfg.

<i>Variable Name</i>	<i>Usage</i>
PAGING	<p>The Paging threshold. The paging rate is obtained by calling a provided REXX function which is an assembler program that uses the GETFLD macro.</p> <p>If server based configuration is being used, the PAGING variable is not required and the PAGING threshold is configured with the PAGING parameter in hobbit-clients.cfg.</p>
SPOOL	<p>The Spool utilization test threshold values. The percentage is obtained by the VSE/POWER D Q command. The SPOOL value only has effect when running in client configuration mode.</p> <p>When running in server configuration mode, the spool utilization thresholds are set with the DISK parameter in hobbit-clients.cfg</p>
GETVIS	<p>GETVIS utilization test partition identifiers. The value SVA can also be coded to test the Shared Virtual Area, specify additional partitions as needed. The utilization thresholds are specified on the server.</p>
PROCS	<p>The running processes test. The name is taken from a similar test on Linux systems, on VSE this is equivalent to running jobs.</p> <p>In client configuration mode, the jobnames to be monitored are listed with the PROCS keyword in HOBVARS.Z. The list of jobs is checked against the currently running jobs (based on D RDR,CDISP=*). If a job in the list is not running a red status message is sent to the Xymon server. An optional number can be provided, this number is checked against the total number of running jobs. If the number is exceeded, a yellow status is returned to the Xymon server.</p> <p>In server configuration mode, the PROCS variables are not required and the jobs to be monitored are specified using the PROC keyword in hobbit-client.cfg,</p>
DISK	<p>Provides values for running the disk utilization test. This test will provide utilization values for VSAM Catalogs, disk volumes and VSE Libraries.</p> <p>In client configuration mode, the default threshold value is specified with the DEF keyword. The list of the items that the disk utilization REXX procedure performs it's test against should follow.</p> <p>In server configuration mode, the DISK configuration on VSE is still required, so that the client knows what to test, but the thresholds are configured on the server.</p>

Figure 4 – Description of HOBVARS client configuration options

Figure 5 shows an example from hobbit-clients.cfg for server side configuration of the monitoring thresholds:

```
HOST=vse1
      CICS      CICSICCF DSA 80 90 EDSA 90 95
```

```

DISK      /VSELIB/PRD2  97 99
DISK      /POWER/DFILE  90 95
GETVIS    F2 80 90 90 95
LOAD      80.0 90.0
VSIZE     80 90
NPARTS    80 95
PAGING    5 10
PROC      %.* 1 25 yellow TRACK=Alljobs "TEXT=All Running Jobs"
PROC      CICS1 1 1 red "TEXT=CICS1 - Production CICS"

```

Figure 5 – Client configuration for z/VSE

The *CICS* keyword defines the yellow and red DSA and EDSA thresholds for a specific CICS application. Code one line for each CICS application that is being monitored.

The *LOAD* values define the thresholds for CPU Utilization, yellow is set to 80%, red is set to 90%.

The *PAGING* values define the thresholds for Paging, yellow is set to 5 pages/second, red is set to 10 pages/second.

The *GETVIS* keyword specifies the yellow and red 24-bit and Any getvis threshold values for a specific z/VSE partition. Code one line for each z/VSE partition that is being monitored.

The *DISK* values define the utilization thresholds for the various disk tests. Each disk test specified is represented as a 'mount' point in Linux df format. The values after the path names are the yellow and red thresholds for that entry. The following mounts points are used:

```

/POWER/QFILE      VSE/Power queue file
/POWER/DFILE      VSE/Power data file
/VSELIB/xxxxxxx   A VSE library with the library name xxxxxxxx
/VOL/xxxxxxx      A VSE volume with the volid xxxxxxx
/VSM/xxxxxxx/vvvvvv/n  A VSAM catalog with the fileid xxxxxxxx, on
volume vvvvvv and space number n

```

The *PROC* values define a job to be checked. The first PROC line will track all running jobs, the second line will make sure that a job called CICS1 is running. The TEXT value is the text that appears on the web page to identify the job(s). The numbers are the lower and upper limits to check for. When checking for a single job (like CICS1), the limits can both be set to one, when checking for multiple jobs (as in all jobs), the limits can be set to a value reasonable for your installation. In either event, if the number falls outside of the limits, the test is turned the color indicated. For instance, if CICS1 is not running, the test turns red. If the total number of jobs in the system is over 25, the test turns yellow.

The *VSIZE* keyword specifies the yellow and red threshold values for z/VSE system virtual storage utilization.

The *NPARTS* keywords indicates the yellow and red thresholds of the number of running partitions against the NPARTS IPL startup parameter.

The disk test is performed by two REXX programs, HOBDSKTS is used to actually perform the tests. Since the disk test on VSE may not need to run during each Xymon cycle (which is every 5 minutes), an additional batch job is used to execute the REXX program that runs the test. The batch job is HOBDSKTS which executes the REXX

program HOBDSKTS. It examines the VSAM catalog, VSE library and VSE volume entries specified in HOBVARS. When it's done, it writes the utilization data to a VSE library for the Xymon client to use to report to the Xymon server. The disk test that is run by the Xymon client performs the spool utilization test (every 5 minutes) and reads the output from HOBDSKTS (in the VSE library) and then sends the data to the Xymon server.

A sample of the HOBDSKTS jobstream is distributed with the code. Although the sample uses POWER event scheduling to run the job every hour, any job scheduler can be used to run the job at an interval that is comfortable for the installation. If you feel it is more important to have timely disk utilization information during first and second shift, the parameters of the scheduled job can be adjusted to run every 15 or 30 minutes, then every 2 hours overnight or weekends.

It is important to note that the longer the interval, the less timely will notifications be sent out in the event of a disk area filling up.

When HOBDSKTS is finished with the disk tests, the results are stored in member HOBDSKRS.Z. The disk utilization program executed by the client (HOBVSEDU) reads that member, reformats the data and sends the results to the Xymon server.

The lines in member HOBVARS.Z that contain the DISK keyword are used to tell HOBDSKTS which disk items to test. There should be one item per line. Each item must be preceded by either CAT= to test a VSAM catalog (with the fileid and datasetname of the catalog following), VOL= to test a VSE volume (with the volid following) or LIB= to test a VSE library (with the fileid of the library following).

In client configuration mode, a line containing DEF provides the default threshold values to apply to all disk utilization tests, unless a value is specified on an individual disk utilization test. After the equal sign is the name of the item of that type to test. In the example above, on the line for the PRD2 library the optional threshold values are specified. From left to right they are the yellow and red thresholds that only apply to that item, otherwise the default threshold values apply.

Tailor the XYMONVSE job to run in a proper class for long running jobs. Modify the LIBDEF to set the proper search library for the TCP/IP stack. A sample of the XYMONVSE job is in Figure 5.

```
* $$ JOB JNM=XYMONVSE,CLASS=L,DISP=L
* $$ LST CLASS=Q,DISP=D
// JOB XYMONVSE      Start Xymon VSE Client
// LIBDEF PROC,SEARCH=LOCAL.LIB
// LIBDEF PHASE,SEARCH=(<tcpip.lib>,LOCAL.LIB)
// OPTION SYSPARM='00'
// EXEC REXX=HOBVSEDR,PARM='PRD2.CONFIG SERVER'
/*
```

```
/&  
* $$ EOJ
```

Figure 5 – Xymon client startup job

In this sample LOCAL.LIB is where the REXX procs, assembler programs and COBOL program are cataloged. The first parameter to the driver program, HOBVSEDR, is the sublibrary where the temporary data used by the client is stored. The second parameter is the configuration mode of the client. The configuration mode can be one of two values, 'CLIENT' or 'SERVER'. The 'CLIENT' setting indicates that the client is in local mode, meaning that all threshold settings are made on the client. The 'SERVER' setting indicates that the client is in server mode, meaning that the Xymon server holds all of the threshold settings. Xymon Server V4.3 is required to use SERVER mode.

The Xymon server should be configured to perform at least a basic connection test on the VSE system where the client is installed. If this is not done, the default behavior of Xymon will be to ignore the status messages received by the client.

The Xymon Client for z/VSE tests are designed to run at 5 minute intervals, similar to the default Xymon test cycle.

Operating the Client

Use the POWER PRELEASE command to start the job at IPL time in the POWER startup PROC or other procedure used to bring up other long running jobs.

To shutdown the Xymon client, issue the following command:

```
MSG XYMONVSE,D=HI
```

It also accepts SHUTDOWN, QUIT, EXIT, TERM and STOP. The termination of Xymon can take up to 15 seconds.

Acknowledgments

I would like to thank the following people:

Frank Ramaekers, Jr. of American Income Life Insurance Company for suggestions on implementation and performance improvements.

Thomas Kern, a consultant for the U.S. Department of Energy, for ideas on the z/VM client that were also used in the changes made to the z/VSE client.

Finally, thanks to Henrik Storer for writing the package and for assistance with the C code for the backend to support the clients.