getsysinfo.sh (current version: 9.23, 21.10.2014)

Latest revision of this document is also available on [HP Support Center](#).

getsysinfo.sh is a data collection shell (ksh) script for HP-UX (≥ 10.20) used by Hewlett Packard hardware support for Business Critical Server (BCS).

It collects diagnostic data for **hardware** troubleshooting from a running [HP 9000](#) or [Integrity Server](#).

The script is optimized for small output and reliability even on systems with bad hardware. It not only collects various log files, but also runs some common commands to obtain additional troubleshooting information.

The result is a gzip`ed tar archive that can be viewed with WinZIP, tar/gzip or other tools. The default output filename (can be changed with the -n option) is:

/tmp/sysinfo_hostname_timestamp.tgz

Some of the raw binary log files in the sysinfo.tgz archive need special viewers and can only be analyzed by HP support.

New since Version 9.0 is the built in data collection from Onboard Administrators (option -oa) via telnet. In addition to this getsysinfo.sh can now call the additional data collectors

hpmpl.286.jar - java tool to collect data from iLO Management Processors (MP) : option -mp

sd2collect.pl   - perl script for data collection from Superdome 2 OA: option -sd2

sginfo   -shell script to obtain Service Guard related information.

Note that these additional tools have to be present in the local directory unless you have one of the embedded getsysinfo versions (getsysinfo.sd2.sh, getsysinfo.mp.sh, **getsysinfo.all.sh**) which have one or all tools embedded as a shar archive.

Syntax (run as root !)

ksh ./getsysinfo.sh [options]

default output will be written to /tmp/sysinfo_hostname_timestamp.tgz
Options (no option is needed unless HP support asks to use a specific option)

```
-a   : get all information except -diag (-h -f -san -xp -p -net -d)
-s   : secure, use only non hanging (script hangs) commands
-h   : get all files from /var/tombstones directory instead only those
     : younger than 180 days.Collect SFM history and SAL records
-f   : do not truncate some of the logfiles to 1024 kB
-f "regexp" : filter syslog files with regexp (Example: "inetd|ftpd")
-sh  : make shell archive /tmp/sysinfo_hostname_timestamp.shar (ascii)
-u   : uuencoded tgz file => /tmp/sysinfo_hostname_timestamp.uu (ascii)
-n FN: explicitly set output filename to FN
-d   : bdf / diskinfo from all disks (scsimgr get_info on 11.31)
-san : capture fcddiag (or td-,fclp-,fcocdiag)
-net : capture netfmt /var/adm/nettl.LOG000
-p   : include information from print_manifest
-diag: capture additional data for diagnostics SW troubleshooting
-diag -diag: capture more diag data - Huge Output may be the result !!!
-sas : force running of sasmgr
-sw  : capture additional information needed for software support
-c [all|N] : run crashinfo on last N or all crashdumps (<180 days)
-xp  : get information about XP diskarrays with xpinfo (may take long)
-oa  : capture show all etc. from onboard administrator (OA) via telnet
-x file : capture file as additional information
```

```
-x 'exec:command' : log the output of command to file addcmds.log
```

Only if external script hpmpl.jar is in same dir or with getsysinfo.mp.sh or getsysinfo.all.sh:
```
-mp  : run external mp collection java program hpmpl.286.jar
-mpopt "options" - set additional options for hpmpl.286.jar
```

Only if external script sd2collect.pl is in same dir or with getsysinfo.sd2.sh or getsysinfo.all.sh:
```
-sd2 : capture Superdome 2 or C[37]000 OA logs with external sd2collect.pl
-sd2opt "options" - set additional options for sd2collect.pl
```

Only if external script sd2collect.pl is in same dir or with getsysinfo.sd2.sh or getsysinfo.all.sh:
```
-sginfo : capture Service Guard logs with external script sginfo
```

The option -s prevents "hanging" commands like ioscan -f, diskinfo, vgdisplay etc.
(hanging means, the command may hang due to a hardware problem, NOT the server !)

Frequently Asked Questions

Why should I use the script ?

It simply saves time if the support engineer gets all information from the OS needed for troubleshooting in one step. And running a script is much more easy than collecting everything manually.

What options should I use ?

The script should be run **as root with ksh** without any option by default. It then collects all data necessary to troubleshoot recent hardware problems.

`ksh getsysinfo.sh`

Note: The script is only tested with Korn Shell (/usr/bin/ksh) and also works with the Posix default shell for the root user (/sbin/sh). Other shells like csh or bash may cause problems.

To also capture MP or OA data (C3000,C7000, Superdome 2) via network:

ksh getsysinfo.all.sh -mp

(java >1.5 needed, only for getsysinfo.mp.sh/getsysinfo.all.sh or if hpmpl.jar is in the same dir)

If one of the commands in the script hangs (usually ioscan), you can abort the current script run (CTRL-C) and run

```
ksh getsysinfo.sh -s
```

Only if you want to collect more data than usual, you can add the options

"-h" (also logfiles files older than 180 days are collected)

"-f" (do not truncate large files to 1 MB)

"-sw" capture also software related information (crashdumps, sginfo etc.)

"-a" capture all information except -diag. Includes `-h -f -d -net -sw -san -xp -p`

The "-diag" option should only be used on special requests, when software problems with STM, EMS or System Fault Manager have to be troubleshot. The output may be huge !

How long does a script run take ?

Of course this depends on the size of the system. Normally it should only take some minutes. But in some cases an ioscan on a full size Superdome can already take 20 minutes …

You can stop the script at any time by pressing CTRL-C and then restart it with the "-s" option which should run much faster (but collects less information).

How big is the output ?

Usually between 200 kB and 1 MB depending on the size of the system. If the -f or -diag options are set, the output may be extremely huge depending on the size of the log files on your system.

Can the script crash the server ?

The script is optimized not to harm the system even in case of hardware problems. Some intelligence is built in to avoid known problems with commands like cstm, sasmgr or fcmsutil.

It only happened in very rare cases that one of the used commands triggered a crash on systems with a severe hardware or software problem.

How can I give feedback to improve the script or report problems ?

Send a mail to getsysinfo@groups.hp.com or give feedback to the HP support engineer who sent you this script. It would be nice to attach the getsysinfo.log file obtain by:

```
ksh getsysinfo.sh 2>&1 | tee getsysinfo.log
```

What is NOT collected by the script ?

- /etc/passwd or any other password or secret containing files are NOT collected

- MP /iLO logs not collected unless you use the "-mp" option and the external hpmpl.jar

- user or application data is not collected

What is collected ?

The /tmp/sysinfo_hostname_date.[shar|tgz|uu] file contains the following Files:

bdf_swapinfo.txt	Output of bdf and swapinfo -tam
bootconf.txt	lvlnboot -v, lifls of bootdisks, AUTO file(s), setboot, ll /stand/vmunix/*
ccerrlog	GSP Chassis codes (from /var/stm/logs/os, binary file, only PA-Risc Servers)
config	general information (uname, model, etc.)
kernelconf.txt	kernel configuration
config.lvm	LVM relevant info (lvmtab, fstab, vgdisplay & lvlnboot -v (not with -s))
crash.txt	information about crash config and /var/adm/crash directory
crashinfo.crash.X.html	Wiith option -c, if /opt/sfm/tools/crashinfo* is present, and a dump is in /var/adm/crash, the dump is analyzed with crashinfo -H -c
cstm.info	STM information logout from all devices.
cprop.txt	output of System Management Homepage related information (*cprop -details -a*)
diagdata.tar	Information needed for diag software troubleshooting (only with -diag option)
diaginfo.txt	Several diagnostic related information. ll -R /var/stm /var/psb /var/opt/sfm /var/opt/resmon.
diskinfo.txt	output of diskinfo -v of all disks (only with -d)
dmesg.txt	Dmesg
event.log*	/var/opt/resmon/log/event.log* files younger than 90 days
[fcd|fclp|fcoc]diag_date.txt.gz	-san: Output of fclpdiag or fcddiag or fcocdiag if fibre channel cards exist for this driver
fcmsutil.out	nice grep output from fcmsutil (Fibre Channel Infos)+ tdlist + fcdlist
fpl.log.??	Chassis Error Log Files on newer systems (from /var/stm/logs/os, younger than 90 days)
getsysinfo.log	log of STDOUT and STDERR of the getsysinfo.sh run
hpvmconsole.vmname.log	Console log of virtual machines captured on the VSP – not if -s is set
hpvmstatus.txt	information about HP virtual machines (and vpars >=6.0)
icod.txt icod.log	output of icod_stat, icod_stat -s (ver≥6) or icod_stat -u (<6)
ioscan.txt	*ioscan -fn (-s: ioscan -fnk).* For HPUX 11.31:*ioscan -fN* and also *ioscan -m dsf, ioscan -m lun , ioscan -m hwpath.* For Superdome 2: *ioscan -m resourcepath*
ioscan_fnk.txt	only for HPUX 11.31: output of *ioscan -fnk* while ioscan.txt contains output of ioscan -fN
irdiag.out	output of *irdiag-v* for Raid4Si cards
lanconfig	*lanscan, nwmgr, lanadmin, netstat outputs* and *netconf* with -net: additional files for hp_apa config
lastlogs.raw	current raw logfile from the online diagnostics (has to be decoded with logtool/stm)
machinfo.txt	output of *machinfo* (IA64 system information)
memlog.txt	Memory error log file (output of logtool commands *vd* and *vda*)
mpt.out	U320 SCSI interfaces: output of *mptutil* and *mptconfig*
nettl.txt	output of *netfmt -f /var/adm/nettl.LOG000* (only with -net option)
olar.txt	Output of *rad -q* or *olrad -q*
oselogs.tgz	raw 11.31 diagnostic logfiles from /var/opt/psb/oselogs
partition.txt	*parstatus, parstatus -c CELL -V, vparstatus, vparstatus -v, vparstatus -e*
ps_top.txt	output of ps -ef and top
rc.log rc.log.old	current and last boot logfile (/etc/rc.log)

rst.log	RST event.log file /var/opt/resmon/log/rst.log
sas.txt	sasdiag like information about SAS disks and cards connected to LSI SAS HBAs (improved own script)
sautil.txt	information about Smartarray raid controllers
service_guard.txt	cmclview -v output (if Service Guard cmcld is running)
sfmconfig.txt	output of *sfmconfig -w -q* (system fault manager config)
sfm_event.log	/var/opt/sfm/log/event.log
sfmlog.txt	system fault manager logs similar to EMS event.log
shutdownlog	/etc/shutdownlog
spmgr.out	EVA diskarray infos
sw_inventory.xml	used by mission critical HP support for patch analysis
swlist.txt	output of *swlist -l bundle*, *swlist -l product* and *swlist -l fileset -a state | grep inst*
syslog.log OLDsyslog.log	(last 2048 MB each) use –f option to capture the complete files
system.krs	krs information from /stand/krs/system.krs contains f.e. SAS dynamic hardware path assignments. (only collected if SAS is used)
tddiag_date.txt.gz	-san: Output of tddiag if fibre channel cards exist for this driver
tombstones	Directory with tombstone files (HPMC or MCA). Also: *crashinfo* outputs and EFI binary MCA files.
tombstones/sfm_sal	Directory with binary CPEs and CMCs from *logExtract* (comes with SysFltMgr)
vmunix.nm.gz	gzipped output of nm /stand/vmunix. Needed for software support for extended analysis of MCAs when no creash dump is written.
vxvm.txt	Information about Veritas Volumemanager (VXVM) config
xpinfo.out	output of xpinfo (for XP diskarrays)

How to capture additional data from an MP or OA with hpmpl.jar ?

At least java 1.5 is needed for this feature. If you do not have getsysinfo.mp.sh or getsysinfo.all.sh (which have the hpmpl tool embedded), the hpmpl.286.jar file needs to be in the same directory as getsysinfo.sh.

Note: getsysinfo.sh is trying to guess the MP IP or OA address with the cprop utility. If successful, it will connect automatically to this IP.

```
getsysinfo.sh -mp       #getsysinfo will ask for login data to MP or OA
getsysinfo.sh -mp -mpopt "-ssh"      #access MP via ssh (default for i2, i4 server)
getsysinfo.sh -mp -mpopt "-h hostname -u user -p password"
```

IMPORTANT: hpmpl will ask for login and password to MP or OA during the getsysinfo.sh run !!!

Note: hpmpl also supports capturing logs via the Onboard Administrator (OA) of C-Class Blade enclosures **and Superdome 2 Servers.**
It will automatically (if no -mpopt option is set) detect if an enclosure is present and by default connect via ssh to the OA instead of the iLO. For i2 and i4 blades, it will then try capturing the MP logs from the Onboard Administrator.
Enhanced example (give OA IP and login when hpmpl asks):

```
getsysinfo.sh -mp -mpopt "-ssh -oamp "  #capture OA+ iLO-MP data from blades and C7000 OA via ssh
```
without -oamp, only "show all" is captured.

To capture only MP logs data from blades in bays 1 and 2 via the OA:

```
getsysinfo.sh -mp -mpopt "-ssh -oamp 12"
```

List of all HPMPL options for use with -mpopt

getsysinfo.sh -mp -mpopt "<list of options>"

HPMPL Options:

-h host name	set hostname
-u username	set username
-p password	set password
-f outputfile	set output filename
-zip	create ZIP archive instead of flat text file
-port port	Port used for telnet (default 23) or ssh (default 22)
-t timeout	timeout in seconds (in case a transaction would time out with the default timeout)
-nocl	disable capture of all console logs
-nosl	disable capture of all event/chassis logs
-nocm	disable commands in cm> mode (for MPs)
-nocr	Disable checkrevs collection for SD2
-maxfpl N	max number of FPL logs to capture (50000)
-maxmca N	max number of MCA logs to capture for SD2 (5)
-oamp [bays]	collect MP logs via Onboard Administrator (OA) bays is by default 12345678 (all blade bays)
-oavc [user pass]	collect VC data via OA (login req.)
-vcdumpurl FTPURL	collect VC connect support dump and save it to the FTP server specified by the FTPURL. FTPURL must be of the form *ftp://user:pass@IP* . No subdir or filename can be specified on the FTP server.
-oadiag	capture additional diagnostic data for OA debug (++diag--, DBGLOG, DBGERR, N_STATS)
-repeat N	run log capture N times
-sleep N	wait N seconds (default 86400) between captures
-zdt	run zdtcrt and zdtpmt on iLO3 to get post mortem traces
-v	print debug info. -v -v logs all incoming input.

How to capture additional data from C-Class or Superdome 2 OA ?

With -mp option (see previous chapter on page 5)

ksh getsysinfo.all.sh -mp

With built in -oa option (no hpmpl.jar or sd2collect.pl needed)

Getsysinfo.sh can capture the "show all" output from Onboard Administrators of C7000, C3000 and Superdome 2 Onboard Administrators by connecting with the HP-UX telnet to the OA. You see the whole output from the telnet session. In case of problems, press CTRL-C.

The output will go into a file show_all.hostname.txt in the sysinfo.tgz .

With sd2collect.pl

With the "-sd2" option, getsysinfo.sh automatically calls the perl script sd2collect.pl which have to be in the same directory as getsysinfo.sh unless you use the embedded versions getsysinfo.sd2.sh or getsysinfo.ms.sh.

The output will go into a subdirectory sd2collect in the sysinfo.tgz archive.

```
getsysinfo.sh -sd2     #sd2collect.pl will ask for login data to OA
getsysinfo.sh -sd2 -sd2opt "-s"   # get less data to save time
getsysinfo.sh -sd2 -sd2opt "-a"    # get all data (output may be huge)
```

```
getsysinfo.sh -sd2 -sd2opt "-d MM/DD/YY" # get only data from after that date
getsysinfo.sh -sd2 -sd2opt "hostname user password"
```

Note: by default, sd2collect only captures data from the last 30 days unless getsysinfo.sh option -h or -a was given.

How to capture Service Guard related information ?

By giving either the -sw or -sginfo option the script sginfo is run.

(see https://h20628.www2.hp.com/km-ext/content-webapp/document?docId=emr_na-c03661098)

It -sginfo was given, the output filename defaults to sysinfo+sginfo_host_date.tgz.

Without that option, getsysinfo.sh still collects by default the cmclview -v information into the file service_guard.txt

How to automatically run crashinfo ?

```
ksh getsysinfo.sh -c          #runs /opt/sfm/tools/crashinfo on the latest dump if present
```

Since version 9.18 getsysinfo.sh can call the crashinfo tool which comes with System Fault Manager and usually is located in /opt/sfm/tools.
You can also specify another crashinfo location with the CRASHINFO environment variable, for example :
```
CRASHINFO=/tmp/crashinfo ksh getsysinfo.sh -c
```

By default, only the last dump is analyzed. To analyze all dumps younger than 6 months you can use:

```
ksh getsysinfo.sh -c all
```

Attention: Analyzing dumps may be a time and storage (/var/adm/crash) consuming task depending on the size of the dump.

Other Examples are:

```
ksh getsysinfo.sh -sw          #-sw for software collections also collects dump
ksh getsysinfo.sh -sw -c 0     #disable dump collection when -sw is set
ksh getsysinfo.sh -c 2         #collect the last two dumps yunger than 6 months
ksh getsysinfo.sh -c all -h    #collect really all dumps (also older 180 days)
```

How can I modify the temporary directory ?

Simply set the environment variable TMPDIR which is used by mktemp to specify the location of temporary files and directories.

For example:

```
TMPDIR=/var/tmp ksh getsysinfo.sh
```

How to run/collect additional commands/files with the -x option ?

Sometimes other information may be needed than collected by getsysinfo.sh. Instead of modifying the script (see next chapter) the -x option can be used to collect additional files or to run additional commands. Of course you should use this option with caution.

-x <file(s)> to capture additional file(s)

By specifying one or more files (full path needed) behind the -x option you can force getsysinfo.sh to capture them as well (output filename will be addfile_<pathname>, all "/" are replaced by "_").

Examples:

```
ksh getsysinfo.sh -x /var/adm/syslog/mail.log
=> addfile__var_adm_syslog_mail.log
ksh getsysinfo.sh -x /var/adm/sw/sw*.log
```

-x 'exec:command' to run additional commands

By putting an "exec:" before the -x argument getsysinfo.sh will run the command and save the output to addcmds.log.

ATTENTION: Make sure you only use safe commands for that and it is highly recommended putting the command into single quotation marks to avoid any shell replacements with the command you intend !!! Better do not use wild-cards – it is risky.

Examples:

```
getsysinfo.sh -x 'exec: ls -lR /var/adm/sw'

getsysinfo.sh -x 'exec:/opt/fcms/bin/fcdutil' -x 'exec:/usr/sbin/sar -d 5 5'
```

Can I modify the script ?

It is safe to modify the parameters at the beginning of the script:

```
#!/usr/bin/ksh
# @(#)Revision: 9.23$ $by S. Stechemesser, HP$ $Date: 10/22/2014$
#
#the below construct is needed no line in getsysinfo.sh is longer than 80 chars
#for cut&paste
READMELINK="https://h20565.www2.hp.com/portal/site/hpsc/template.PAGE/public"
READMELINK="$READMELINK/kb/docDisplay/?docId=emr_na-c03800758"
export LANG=C
GETSYSINFOVERSION="9.23"
#CKSUM: 3194520570
#use the TMPDIR environment variable if needed - see man mktemp
TMPDIR=${TMPDIR:-/tmp} # set TMPDIR to /tmp if not set
NAMEBASE=sysinfo  #default output name prefix
#add hostname + timestamp by default
NAME=${NAMEBASE}_`hostname`_`date +%Y%m%d%H%M`
TDIR=$TMPDIR/$NAME
MPJAVA=0           #do not run jar file by default, only with -mp option
GETOA=0            #get OA info via telnet
SD2=0              #do not run be default
SGINFO=0           #do not run sginfo by default
MAXFILESIZE=1024 # Maximum size of some logfile in kbytes
GETFIRSTLINES=2000 # if logfiles are truncated, also the first lines are saved
MAXBACKDATE=180 # for some logfiles: get information less than 180 days old
RUNCRASHINFO=0 #1=run crashinfo on last crash
               #2=run crashinfo for all crashes
SAN=0 #do not capture fcddiag by default
NET=0 #do not capture nettl and netstat logs by default
diskjn=0 #do not query all disks with diskinfo or scsimgr by default
INITDATA=0 # do not capture initdata stuff by default
PMANI=0 # do not use print_manifest by default
SASOPTION=1 # 0: run sasmgr in any case
XPINFO=0 #1: run XPINFO
NOHIST=0 #evweb -b history by default (set to 1 if it makes problems)
NMVMUNIX=0 # do not collect symbols from vmunix by default
ESCSIDIAG=0 #do not capture escsi_diag output
SWINFO=0   #higher values capture more data for SW suport
OSELOGS=1   #capturing of /var/opt/psb/oselogs
#
```

```
#better do not change parameters below here unless you know what you are doing
#Integrity check
```

Note: when the script is modified and the **CKSUM:** line is not adjusted properly,   a warning about the modification will be printed and the script will pause for 15 seconds when started.

How to check if I have the official getsysinfo.sh script or a modified ?

Simply run "cksum getsysinfo.sh" and verify with the below values.

The cksum outputs of getsysinfo.sh version 9.23 official revisions are:

```
2176148563      54116       getsysinfo.sh
3089166024      230078      getsysinfo.all.sh
164757447       88606       getsysinfo.sd2.sh
4145209390      170673      getsysinfo.mp.sh
351488249       205162      getsysinfo.ms.sh
```

How to transfer data via ASCII terminal applications with cut&paste ?

getsysinfo.sh can be transferred via cut&paste to the target server via a terminal application (like putty). No line is longer than 80 chars.

```
cat > /tmp/getsysinfo.sh
#then paste the script into your terminal application
CTRL-D
```

You can use the "-u" option to transfer the result back to you PC.

```
ksh /tmp/getsysinfo.sh -u
#now start logging to a file on your terminal application
cat /tmp/sysinfo.uu
#stop logging
```

You can then unpack the file on a Unix Server by deleting everything before "begin" with

```
uudecode logfile
```

and you have a gzipped tar archive that can be unpacked with WinZIP or gunzip and tar.

Can I run the script regularly from cron ?

This is not recommended.

Although the script is not known to cause problems, it was not written for collecting configuration data but to troubleshoot hardware problems. A lot of commands are used which directly interact with internal and external hardware, LVM and other system resources which may cause strange logs in syslog.log or device logs if running too often.