

Date: Thu, 12 Oct 1995 13:48:24 -0500

crashsystem(1m)

crashsystem(1m)

NAME

crashsystem - stores raw crashdump data for each hypernode on the designated partition

SYNOPSIS

/spp/bin/crashsystem all

DESCRIPTION

crashsystem is executed from the SPP test station to store crashdump data for all hypernodes, each on its designated partition.

SYSTEM SETUP

You must have a crashdump partition for each node you wish to dump. To see if a node already has one, run /etc/crashutil -p on your SPP. There should be a list of nodes and assigned disk partitions. If no crash partition exists for a given node, you must create one.

To create a crash partition, first pick a disk on the desired node. You can determine on which node a disk resides using the diskutil command 'show disks'. Then once an appropriate disk is located, use 'set disk <name>' and Next, find or create a partition of the proper size. A one hypernode system should only require 40MB, while a two node system may require as much as 70MB, and larger systems 128MB. The amount of space needed depends on the amount of global shared memory on the SPP and the rate of compression achieved. Then, each partition must have the crashdump flag set. Setting this flag for the fourth partition 'd' would be done with the diskutil command 'set part d flag c'. Lastly, if you added or modified any crashdump partitions, you must reboot your SPP for the OS to recognize them.

If you are getting an 'Unknown argument' message on the console, check to make sure you are running a SCSI ROM V5.0 or greater. Versions of SPP OS and OBP previous to 3.1 may not have the crashdump feature. If you are running old versions, check your System Administrator's Guide to see if it is supported.

The default crashdump tunables are set up to dump compressed output of a crashreport, kernel and server for each node. If you wish to change the default behavior, see your System Administrator's Guide.

EXAMPLE

```
test station % crashsystem all
crashsystem: (version) (build date)
Running this program will crash the SPP system
Do you want to continue? [y/n] y
```

Convex Computer Corporation

- 1 -

SPP-UX Operating System

crashsystem(1m)

crashsystem(1m)

NOTE: When this command completes, the crashdump has NOT completed. The completion of crashsystem merely indicates that crashdump has been initiated on all nodes (or the node specified). Crashdump has completed when an 'ok' prompt appears on the system console.

To check the current status of a crashdump, you can spy on the crashdump virtual consoles. The output from crashdump on each node is directed to a different virtual console. Virtual consoles can be accessed from the test station with the 'sn_cns1' command. The crashdump virtual consoles start at 2090 for node 0 and increment by one for each node. To look at the crashdump output for node 1, you would execute the following command :

```
test_station % sn_cns1 -S 2091
```

This will tell you whether the crashdump is in crashreport, kerneldump, serverdump, waiting for other nodes to complete, or concluding by entering OBP. Monitoring the crashdump virtual consoles is not necessary for crashdump to work, only for debugging purposes. Once all nodes have entered OBP, you may use the system console ('sn_cns1 -F 1') and type 'boot' at the 'ok' prompt to boot your system in the usual manner.

SPP crashdump is very different from C-series crashdump. Taking a crashdump on SPP usually takes only a few minutes. Even if the crashdump fails, you will know this and can proceed to reboot.

RELATED COMMANDS

When a crashdump is taken, it will overwrite all crashdump data from the previous crashdump. It is important to run 'crashutil' on the SPP to retrieve crashdump data from the crashdump partition before taking another crashdump (see crashutil.lm). Due to the large amount of space they consume, crash files are compressed by default.

```
spp % crashutil -p
Current crashdump partitions: last date written
node 0 : sd0e (date)
spp % crashutil -i sd0e -o crashfile.0
gzip command: '/usr/contrib/bin/gzip > crashfile.0.z'
Converting kernel address ranges
Converting server address ranges
```

These files may then be referred to your TAC, a developer debugging the SPP OS problem, or by an experienced gdb user. To read a crashdump you must have a version of gdb linked with the libcras

h

which matches your kernel. The output file is usable directly by gdb if the target 'crash' and the name of the crash file are specified. Then to debug the kernel, you would specify 'attach 0' to attach to the kernels address space. To attach to the server, type 'attach -3'. We have also extended gdb with the commands 'info crash' and 'info hw'.

Convex Computer Corporation

- 2 -

SPP-UX Operating System

crashsystem(lm)

crashsystem(lm)

```
spp % gunzip crashfile.0.z
spp % gdb /scratch/mach_kernel
GDB is free software and you are welcome to distribute copies of i
under certain conditions; type "show copying" to see the condition
```

t

s.

There is absolutely no warranty for GDB; type "show warranty" for details.

```
GDB 4.10-11/03/94 (hppa1.1-hp-hpux),
Copyright 1993 Free Software Foundation, Inc...
(gdb) target crash crashfile.0
#0 update_priority (thread=0x2da500) at ../../kern/sched_prim.c:1
311
    ../../kern/sched_prim.c:1311: No such file or directory.
(gdb) attach 0
Attaching program:
/scratch/mach_kernel task 0
You can't do that without a process to debug
(gdb)
```

Note that the message 'You can't do that without a process to debug' is normal and should be ignored.

RESTRICTIONS

Since crashdump sends a HPMC to gain control on the SPP processors, anything which would prevent a processor from receiving that HPMC will also stop crashdump from starting. For instance, if you can't read/write SPP memory from the test station, the crashdump will not work. Likewise, if a SPP processor is already stuck inside a HPMC, that crashdump might not work.

If a processor is hung due to network problems or space allocation problems, it may not be able to open the proper virtual console. In this case, the output for that virtual console is routed to the system console.

If you do not have a crash partition configured for a node, or do not reboot after a reconfiguration, the crashdump for that node will not work.

KNOWN PROBLEMS

Crashdump would sometimes hang while doing a dump of the server. The last line seen on the crashdump virtual console would be 'In serverdump'. This problem has been fixed since OOW_3_0_29_1.

Crashdump would sometimes hang after printing the 'Redirecting crashdump output to virtual console ...' message on the console when one or more of the processors were uninterruptible. As of OOW_3_0_29_1, crashdump will timeout after waiting for 10 seconds for all processors to respond.

Crashdumps would sometimes dump much more data than it needed to. This would cause the crashdump partition to be filled and the crashdump

Convex Computer Corporation

- 3 -

SPP-UX Operating System

crashsystem(1m)

crashsystem(1m)

would fail with a 'disk partition filled' message. This has been fixed in 3.0.4, and may be worked around by using larger partitions.

When a crashdump successfully completes, it prints out the message "Panic OBP: calling obp_enter". While this message seems to indicate an error, it is normal. It will be removed in a future OS to help avoid confusion.

Crashdump will not always work on nodes built with fewer than 8 CPUs.

One node reaching the OBP lock prematurely is usually indicative of missing or misconfigured crash partitions.

Nodes that do not report by the ten-second timeout will probably not fill their crashdump partitions.

REPORTING NEW PROBLEMS

We often get Problem Reports concerning OS panics or asserts with the backtrace that was printed on the console. Unfortunately, this is often insufficient for debugging the root cause of the problem. In many cases, we have a much better chance of resolving the problem if we can debug the system while it is down or have a crashdump to look at. Since it is usually not convenient for the customer to leave their system down for extended periods of time, a crashdump is the next best thing. If you have a problem taking a crashdump please submit a contact report. Include a tail of the text in the crashdump virtual consoles, the system console, and a description of what triggered the crashdump.

Date: Thu, 12 Oct 1995 13:46:58 -0500

```
# $CHheader: tunables.cnx_mpp1 1.19 1995/05/04 14:55:47 $
#
# Copyright 1993, CONVEX Computer Corporation.
# This document is copyrighted. All rights are reserved. This
# document may not, in whole or part, be copied, duplicated,
# reproduced, translated, electronically stored or reduced to
# machine readable form without prior written consent from CONVEX
# Computer Corporation.
#####
# There are many tunables that are not in this file. They take their
# default values when the system boots. For more information about
# tunables, please see the online tunables manual page.
#
# Command: man tunables
#####
# Knowledge of the internal operations of SPP-UX is recommended before
# altering these values. The following formats for values are accepted -
# s string
# n decimal numeric
# 0xn hex numeric
# nK n * 1024
# nM n * 1024 * 1024
#
# Where ranges are present in the key, such as "1[0..15]", this generally
# refers to the values on each of the possible nodes in the system. With
# this format all entries in the array range are set to the single value
# specified.
#
#
# Microkernel tunables
#
# Event Logger internal buffer, for holding events before syslog becomes active
Event Logger,buffer_size:desc=Event Logger Buffer Size:l=64k:default=64k:
Event Logger,event logger threshold:desc=Event Logger Condition Code Threshold:l
=0:default=0:
Event Logger,event logger boot threshold:desc=Event Logger Condition Code Boot T
hreshold:l=2:default=2:
# Control the LCD processor heartbeat update (0:Off, 1:On)
LCD Heart Beat,control:desc=Control for LCD update:l=1:default=1:
#
# Server tunables
#
# Percent of memory dedicated to the filesystem buffer cache
Fileserver,buffer_cache_percent:l[0..15]=10:
# Set wired device buffer size for raw disk devices
Fileserver,disk_wdb_size:l[0..15]=128k:
# Set wired device buffer size for tape devices
Fileserver,tape_wdb_size:l[0..15]=512k:
# Percent of disk space needed to be free to resume accounting
Server,acctresume:desc=Percent of disk needed to resume accounting:l=4:
# Percent of disk space at which pint accounting is suspended
Server,acctsuspend:desc=Percent of disk when accounting is suspended:l=2:
# Default size of data segment
Server,dfldsiz:desc=Default Data Size Limit:l=512M:
# Have all servers panic gracefully when any server panics
Server,distribute_panic:l[0..15]=1:
# Set DST
```

```
Server,dst:desc=use dst rule:1=1:

# Internet Checksum Method (old=1, new=2, both=3)
Server,incksum:desc=Internet Checksum Method:1=2:

# Maximum number of child processes allowed
Server,maxuprc:desc=Maximum number of child processes:1=500:

# Maximum size of data segment adjustable via rlimit
Server,maxdsiz:desc=Maximum Data Size Limit:1=1984M:

# Maximum stack size adjustable via rlimit
Server,maxssiz:desc=Maximum Stack Size Limit:1=512M:

# Tune system tables for this many users
Server,maxusers:desc=Tune system tables:1=384:

# Tune the number of file locks
Server,nflocks:desc=number of file locks:1=200:

# Tune the number of ptys
Server,npty:desc=specify the number of ptys:1=60:

# Print the server tunable values during boot
Server,printtunables:desc=print tunable values:1=0:

# Set the default timezone
Server,timezone:desc=Default timezone:1=300:

# Enable unique core file names
Server,unique_core_names:desc=Enable unique core file names:1=0:

Server,semnmi:1[0..15]=512:

Server,semnms:1[0..15]=1024:

Server,shmmax:1[0..15]=1024M:

Server,shmmni:1[0..15]=512:

Server,shmseg:1[0..15]=256:
```

Date: Thu, 12 Oct 1995 13:30:37 -0500

Where <list> consists of any number of the following:

Adding a + to one of the elements in the list will invert the sort. ex. 'pot -o +%CPU' puts most cpu intense at bottom

PID - pid this info is for
 UTIME - user time accrued
 STIME - system time accrued
 UID - effective uid of owner
 USERNAME - name of owner
 GID - effective gid of group
 PPID - parent process id
 PGRP - process group
 STATE - process type (act/zomb)
 SFLAG - process run status
 COMMAND - real command name
 RUID - real uid of process
 RUSERNAME - real name of owner
 SVUID - saved uid of process
 SVUSERNAME - saved name of owner
 RGID - real gid of process
 SVGID - saved gid of process
 SID - session id
 TPGRP - terminal process group
 TSID - terminal session id
 JOBC - number of procs in pgrp
 CURSIG - current signal
 PND SIG - pending signals
 SIGMSK - current signal mask
 IGNSIG - ignored signals
 CTHSIG - caught signals
 DSIZE - size of data segment
 TSIZE - size of text segment
 TT - tty owning process
 NI - niceness of the process
 BTIME - time the process started
 PRI - priority of process
 TUTIME - user time accrued/thread
 TSTIME - system time accrued/thread
 TID - Thread id
 SSIZE - size of stack segment
 MSSIZ - size of stack segment
 SCID - Sub Complex thread is running in
 NODE - Node thread is running on
 PNODE - Physical node process is running on
 SCNAME - Name of SC process is running on
 LONGCMD - Command line with args
 SIZE - total size of process
 %CPU - percentage of the cpu
 TIME - total time accrued for thread
 PTIME - total time accrued for process

Default -F TT PID USERNAME PRI NI SIZE STATE TIME %CPU COMMAND
 Default -O %CPU PRI

Date: Thu, 12 Oct 1995 13:29:22 -0500

sysinfo -av AIL library version:
SPP-UX_ail 3.1 L31 09/14/95 (c)Copyright 1995 Convex [CNX_MPP1]

sysinfo -cpu [phys_node [cpu]] cpu information:

| node | cpu | flags | subcomplex |
|------|-----|-------------------------|------------|
| 0 | 0 | Server, Running | System |
| 0 | 1 | Master, Server, Running | System |
| 0 | 2 | Server, Running | System |
| 0 | 3 | Server, Running | System |
| 0 | 4 | Server, Running | System |
| 0 | 5 | Server, Running | System |
| 0 | 6 | Server, Running | System |
| 0 | 7 | Server, Running | System |
| 1 | 0 | Server, Running | System |
| 1 | 1 | Master, Server, Running | System |
| 1 | 2 | Server, Running | System |
| 1 | 3 | Server, Running | System |
| 1 | 4 | Server, Running | System |
| 1 | 5 | Server, Running | System |
| 1 | 6 | Server, Running | System |
| 1 | 7 | Server, Running | System |

sysinfo -i id number of system:

system id: 65622

sysinfo -lc load, complex:

complex load average: 10.66 10.85 13.01

sysinfo -lcn [phys_node] load, complex and nodes:

complex load average: 10.66 10.85 13.01
node 0 load average: 5.21 5.39 6.78
node 1 load average: 5.45 5.46 6.23

sysinfo -ls [sc_name] load, subcomplex:

System load average: 8.84 9.01 11.06

sysinfo -lsn [sc_name [logical_node]] load, subcomplex logical node:

System load average: 8.84 9.01 11.06
logical node 0 load average: 4.26 4.44 5.75
logical node 1 load average: 4.57 4.57 5.32

sysinfo -memc complex memory:

| COMPLEX MEMORY | max | allocated | free |
|----------------|-------|-----------|-------|
| global | 32M | 9M | 23M |
| node private | 1683M | 358M | 1325M |
| buffer cache | 204M | | |
| network cache | 128M | | |
| total | 2048M | 367M | 1348M |

sysinfo -memn [phys_node] node memory:

| NODE 0 MEMORY | max | allocated | free |
|---------------|-------|-----------|------|
| global | 16M | 4M | 11M |
| node private | 841M | 185M | 656M |
| buffer cache | 102M | | |
| network cache | 64M | | |
| total | 1024M | 189M | 667M |
| NODE 1 MEMORY | max | allocated | free |
| global | 16M | 3M | 12M |
| node private | 841M | 172M | 669M |
| buffer cache | 102M | | |
| network cache | 64M | | |
| total | 1024M | 176M | 681M |

sysinfo -msg

System V message info:

msgmax: 8192
msgmnb: 16384
msgmni: 50
msgtql: 40

sysinfo -mv

Mach kernel version:

SPP-UX_mk 08.29.95 L32 N crusader:/src [CNX_MPP1_FAST]

sysinfo -node [phys_node]

node information:

node 0: 1024M memory, 8 cpus: 7, 6, 5, 4, 3, 2, 1, 0
node 1: 1024M memory, 8 cpus: 7, 6, 5, 4, 3, 2, 1, 0

sysinfo -sema

System V semaphore info:

semnmi: 512
semmsl: 500
semopm: 500
semume: 10
semvmx: 32767
semaem: 16384

sysinfo -shmem

System V shared memory info:

shmmax: 1073741824
shmmni: 1
shmmni: 512
shmseg: 256

sysinfo -stat

Unix server statistics:

syscall count: 1507730
fork count: 11424

sysinfo -sv

Unix server version:

SPP-UX_server 08.29.95 L32 N crusader:/src [CNX_MPP1_FAST]