

```

invoke:
  mdb rps core
  mdb -p <pid> # attach to process:
  mdb rps -l <lac_file> <recording>
  UMEM_DEBUG=default UMEM_LOGGING=transaction \
  LD_PRELOAD=libumem.so.1 ./rps

side-note try: `pstack core`

help/info:
  ::help
  ::dcmds
  ::formats
  ::walkers

some basic control:
  ::quit / $q - quit
  ::stack / $c / $C - print current stack trace
  ::walk thread |::findstack

examine memory:
  <addr>/<num><format> - display memory in format

  e.g.
  1025c5a50/10X - show 10 32bit hex integers:
  1025c5a50/s - show as string
  1025c5a50::print [type] - print as type (if type can be
determined (needs debug info in binary))
  0x70a9ae00,5/KKn

  <addr>::dis - disassembly

libumem use: http://access1.sun.com/techarticles/libumem.html
  ::findleaks
  ::umem_status
  ::umem_verify

  ::umem_log
  ::umem_log!wc -l
  ::walk umem_log |::bufctl_audit
  ::walk umem_log |::bufctl_audit ! cat > file
  ::walk umem_alloc_12288 |::bufctl_audit
  0x70ae3200$<bufctl_audit

```

```

::umastat
::umalog
::umausers
::umausers umem_alloc_80
::umausers !cat >umausers-3.log

::walk umem_alloc_96 | ::map '*(.+60+8)' | ::bufctl_audit

::umem_cache ! grep umem_alloc_96
10020c028::walk bufctl | ::bufctl_audit

113429240::whatis

control running process:
  # no symbols available until process loaded so set breakpoint
in main
  ::bp main
  ::run -c -l <lacfile> <recordingfile>
  ::nm !grep UseridCache|grep Add
  ::bp _ZN11UseridCache3AddEjP3RAIP4IMSI
  ::events

brief gdb to mdb list:
http://blogs.sun.com/roller/page/eschrock/20050510

run control:
  :A / [pid]::attach [core|pid]
  :R / ::release -a
  :r / ::run args
  :c / ::cont [sig]
  :e / ::next [sig]
  :s :u / ::step [over|out] [sig]
  $c / [addr]::stack [count]

watch/break-points (events):
  :b / [addr]::bp .. stuff .. [cmd] [-n count] sym
  :a :p :w / addr[,len]::wp
  :t / [signal]::sigbp
  :d / [addr]::delete [id|all]
  $b / ::events [-av]

```

```

e.g.

# if umem causes a core then the first thing to run is ::
umem_status

# finding leaks (attach to process using mdb -p <pid>

> ::findleaks
CACHE          LEAKED          BUFCTL CALLER
00000001025e2b28    2846 00000001026627f0
_Z18DecodeIupsUserplIpPP6CNlPdu+0xac

> 00000001026627f0::bufctl_audit
          ADDR          BUFADDR    TIMESTAMP THR
LASTLOG   CONTENTS
          CACHE          SLAB          NEXT DEPTH
00000001026627f0 0000000102660ba0 64b785c211775a 1
00000001025d98c0 0000000000000000
00000001025e2b28 00000001025fbc30 000000010fe62720 7
                libumem.so.1`umem_cache_alloc+0x228
                libumem.so.1`umem_alloc+0x3c
                libumem.so.1`malloc+0x28
                _Z18DecodeIupsUserplIpPP6CNlPdu+0xac
                _Z13process_blockv+0x6ac
                main+0x13d4
                _start+0x7c

# examining libumem buffer redzones and debug info

::walk umem_alloc_4096
1025b6240+0x1000/20X
10258cab0::bufctl_audit
1025b6240+0x1008/1J | ::bufctl_audit

* <----- mi.malloc_size ---->
* <----- cp.cache_bufsize ----->
* <----- cp.cache_chunksize ----->
* +-----+-----+-----+-----+
* | /tag/// | mallocsz | /round-off///// | /debug info///// |
* +-----+-----+-----+-----+

```

```

# using ::umem_verify

::umem_verify
>> umem_alloc_16          3d6c8 1 corrupt buffer

3d6c8::umem_verify
>> Summary for cache 'umem_alloc_16'
>> buffer 49fe0 (free) seems corrupted, at 0

49fe0/10X
>> 0x49fe0:      deadbeef  deadbeef  30313233  3400beef
>>              feedface  feedface  54780    f4ebb36e
>>              deadbeef  deadbeef

> ::umalog |::grep 49fe0

```

libumem use: <http://access1.sun.com/techarticles/libumem.html>

<http://blogs.sun.com/roller/resources/jwadams/mdb-cheatsheet.pdf>

<http://blogs.sun.com/roller/page/jwadams/20050614>

[Solaris Modular Debugger Guide](#)

[Solaris 10 Software Developer Collection >> Multithreaded Programming Guide >> 7. Compiling and Debugging](#)

[http://blogs.sun.com/roller/page/jwadams?entry=some\\_block\\_comments\\_about\\_libumem](http://blogs.sun.com/roller/page/jwadams?entry=some_block_comments_about_libumem)

mdb-discuss forum:

<http://opensolaris.org/jive/forum.jspa?forumID=4&start=0>

brief gdb to mdb list:

<http://blogs.sun.com/roller/page/eschrock/20050510>

James Coleman  
Software Developer

<http://www.commprove.com/>

