

```

*****
9928 Wed Apr 30 15:43:09 2014
new/usr/src/cmd/mdb/common/modules/genunix/ldi.c
4785 mdb crashes in ::ldi_handle
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
24 * Use is subject to license terms.
25 */
27 #pragma ident      "%Z%M% %I%      %E% SMI"

27 #include <sys/types.h>
28 #include <sys/sysmacros.h>
29 #include <sys/dditypes.h>
30 #include <sys/ddi_impldefs.h>
31 #include <sys/ddi_propdefs.h>
32 #include <sys/modctl.h>
33 #include <sys/file.h>
34 #include <sys/sunldi_impl.h>

36 #include <mdb/mdb_modapi.h>
37 #include <mdb/mdb_ks.h>

39 #include "ldi.h"

41 /*
42  * ldi handle walker structure
43  */
44 typedef struct lh_walk {
45     struct ldi_handle      **hash; /* current bucket pointer */
46     struct ldi_handle      *lhp;  /* ldi handle pointer */
47     size_t                 index;  /* hash table index */
48     struct ldi_handle      buf;    /* buffer used for handle reads */
49 } lh_walk_t;
_____unchanged_portion_omitted_____

361 int
362 ldi_handle(uintptr_t addr, uint_t flags, int argc, const mdb_arg_t *argv)
363 {
364     int             ident = 0;
365     int             refs = 1;

369     if (mdb_getopts(argc, argv,
370         'i', MDB_OPT_SETBITS, TRUE, &ident) != argc)

```

```

371         return (DCMD_USAGE);

373     if (ident)
374         refs = 0;

367     /* Determine if there is an ldi handle address */
368     if (!(flags & DCMD_ADDRSPEC)) {
369         if (mdb_walk_dcmd("ldi_handle", "ldi_handle",
370             argc, argv) == -1) {
371             mdb_warn("can't walk ldi handles");
372             return (DCMD_ERR);
373         }
374         return (DCMD_OK);
375     }

376     if (mdb_getopts(argc, argv,
377         'i', MDB_OPT_SETBITS, TRUE, &ident) != argc)
378         return (DCMD_USAGE);

380     if (ident)
381         refs = 0;

383 #endif /* ! codereview */
384     /* display the header line */
385     if (DCMD_HDRSPEC(flags))
386         ldi_handle_header(refs, ident);

388     /* display the ldi handle */
389     if (ldi_handle_print(addr, ident, refs))
390         return (DCMD_ERR);

392     return (DCMD_OK);
393 }

395 void
396 ldi_ident_help(void)
397 {
398     mdb_printf("Displays an ldi identifier.\n"
399         "Without the address of an \"ldi_ident_t\", \"
400         \"print all identifiers.\n"
401         "With an address, print the specified identifier.\n");
402 }

404 void
405 ldi_handle_help(void)
406 {
407     mdb_printf("Displays an ldi handle.\n"
408         "Without the address of an \"ldi_handle_t\", \"
409         \"print all handles.\n"
410         "With an address, print the specified handle.\n\n"
411         "Switches:\n"
412         "  -i print the module identifier information\n");
413 }

```