

\*\*\*\*\*

36982 Mon Aug 19 12:09:35 2019

new/usr/src/man/man3/Intro.3

11582 3SCF isn't the object-caching memory allocation library

\*\*\*\*\*

```

1  \" te
2  .\" Copyright 2019 Peter Tribble
2  .\" Copyright 2017 Peter Tribble
3  .\" Copyright 2015 Joyent, Inc.
4  .\" Copyright (c) 2009, Sun Microsystems, Inc. All Rights Reserved.
5  .\" The contents of this file are subject to the terms of the Common Development
6  .\" See the License for the specific language governing permissions and limitat
7  .\" the fields enclosed by brackets \"[]\" replaced with your own identifying info
8  .TH INTRO 3 \"Aug 19, 2019\"
8  .TH INTRO 3 \"Nov 26, 2017\"
9  .SH NAME
10 Intro, intro \- introduction to functions and libraries
11 .SH DESCRIPTION
12 .LP
13 This section describes functions found in various Solaris libraries, other than
14 those functions described in Section 2 of this manual that directly invoke UNIX
15 system primitives. Function declarations can be obtained from the
16 \fB#include\fR files indicated on each page. Pages are grouped by library and
17 are identified by the library name (or an abbreviation of the library name)
18 after the section number. Collections of related libraries are grouped into
19 volumes as described below. The first volume contains pages describing the
20 contents of each shared library and each header used by the functions, macros,
21 and external variables described in the remaining volumes.
22 .SS \"Library Interfaces and Headers\"
23 .LP
24 This volume describes the contents of each shared library and each header used
25 by functions, macros, and external variables described in the remaining
26 volumes.
27 .sp
28 .ne 2
29 .na
30 \fB(3LIB)\fR
31 .ad
32 .sp .6
33 .RS 4n
34 The libraries described in this section are implemented as shared objects.
35 .sp
36 Descriptions of shared objects can include a definition of the global symbols
37 that define the shared objects' public interface, for example \fBSUNW_1.1\fR.
38 Other interfaces can exist within the shared object, for example
39 \fBSUNWprivate.1.1\fR. The public interface provides a stable, committed set of
40 symbols for application development. The private interfaces are for internal
41 use only, and could change at any time.
42 .RE
43 .sp
44 .ne 2
45 .na
46 \fB(3HEAD)\fR
47 .ad
48 .sp .6
49 .RS 4n
50 The headers described in this section are used by functions, macros, and
51 external variables. Headers contain function prototypes, definitions of
52 symbolic constants, common structures, preprocessor macros, and defined types.
53 Each function described in the remaining five volumes specifies the headers
54 that an application must include in order to use that function. In most cases
55 only one header is required. These headers are present on an application
56 development system; they do have to be present on the target execution system.
57 .RE

```

58 .SS \"Basic Library Functions\"

61 .LP

59 The functions described in this volume are the core C library functions that  
60 are basic to application development.

61 .sp

62 .ne 2

63 .na

64 \fB(3C)\fR

65 .ad

66 .sp .6

67 .RS 4n

68 These functions, together with those of Section 2, constitute the standard C  
69 library, \fBlibc\fR, which is automatically linked by the C compilation system.  
70 The standard C library is implemented as a shared object, \fBlibc.so\fR. See  
71 \fBlibc(3LIB)\fR and the \"C Compilation System\" chapter of the \fBIANSI C  
72 Programmer's Guide\fR for a discussion. Some functions behave differently in  
73 standard-conforming environments. This behavior is noted on the individual  
74 manual pages. See \fBstandards(5)\fR.

75 .sp

76 The \fBlibpthread\fR and \fBlibthread\fR libraries are filter libraries on  
77 \fBlibc\fR that are used for building multithreaded applications:

78 \fBlibpthread\fR implements the POSIX (see \fBstandards(5)\fR) threads

79 interface, whereas \fBlibthread\fR implements the Solaris threads interface.

80 See \fBMULTITHREADED APPLICATIONS\fR, below.

81 .RE

83 .sp

84 .ne 2

85 .na

86 \fB(3C\_DB)\fR

87 .ad

88 .sp .6

89 .RS 4n

90 These functions constitute the threads debugging library, \fBlibc\_db\fR. This  
91 library is implemented as a shared object, \fBlibc\_db.so\fR, but is not  
92 automatically linked by the C compilation system. Specify \fB-ld\_db\fR on the  
93 \fBcc\fR command line to link with this library. See \fBlibc\_db(3LIB)\fR.

94 .RE

96 .sp

97 .ne 2

98 .na

99 \fB(3MALLOC)\fR

100 .ad

101 .sp .6

102 .RS 4n

103 These functions constitute the various memory allocation libraries:  
104 \fBlibmalloc\fR, \fBlibbsdmalloc\fR, \fBlibmapmalloc\fR, \fBlibmtmalloc\fR, and  
105 \fBlibumem\fR. Each of these libraries is implemented as a shared object  
106 (\fBlibmalloc.so\fR, \fBlibbsdmalloc.so\fR, \fBlibmapmalloc.so\fR,  
107 \fBlibmtmalloc.so\fR, and \fBlibumem.so\fR). These libraries are not  
108 automatically linked by the C compilation system. Specify \fB-lmalloc\fR,  
109 \fB-lbsdmalloc\fR, \fB-lmapmalloc\fR, \fB-lmtmalloc\fR, and \fB-lumem\fR to  
110 link with, respectively, \fBlibmalloc\fR, \fBlibbsdmalloc\fR,  
111 \fBlibmapmalloc\fR, \fBlibmtmalloc\fR, and \fBlibumem\fR. See  
112 \fBlibmalloc(3LIB)\fR, \fBlibbsdmalloc(3LIB)\fR, \fBlibmapmalloc(3LIB)\fR,  
113 \fBlibmtmalloc(3LIB)\fR, and \fBlibumem(3LIB)\fR.

114 .RE

116 .SS \"Networking Library Functions\"

120 .LP

117 The functions described in this volume comprise the various networking

118 libraries.

119 .sp

120 .ne 2

121 .na

```

122 \fB(3COMMPUTIL)\fR
123 .ad
124 .sp .6
125 .RS 4n
126 These functions constitute the communication protocol parser utilities library,
127 \fBlibcommputil\fR. This library is implemented as a shared object,
128 \fBlibcommputil.so\fR, but it is not automatically linked by the C compilation
129 system. Specify \fB-lcommputil\fR on the \fBcc\fR command line to link with
130 this library. See \fBlibcommputil\fR(3LIB).
131 .RE

133 .sp
134 .ne 2
135 .na
136 \fB(3DLPI)\fR
137 .ad
138 .sp .6
139 .RS 4n
140 These functions constitute the data link provider interface library,
141 \fBlibdlpi\fR. This library is implemented as a shared object,
142 \fBlibdlpi.so\fR, but it is not automatically linked by the C compilation
143 system. Specify \fB-ldlpi\fR on the \fBcc\fR command line to link with this
144 library. See \fBlibdlpi\fR(3LIB).
145 .RE

147 .sp
148 .ne 2
149 .na
150 \fB(3DNS_SD)\fR
151 .ad
152 .sp .6
153 .RS 4n
154 These functions constitute the DNS service discovery library, \fBlibdns_sd\fR.
155 This library is implemented as a shared object, \fBlibdns_sd.so\fR, but it is
156 not automatically linked by the C compilation system. Specify \fB-ldns_sd\fR on
157 the \fBcc\fR command line to link with this library. See \fBlibdns_sd\fR(3LIB).
158 .RE

160 .sp
161 .ne 2
162 .na
163 \fB(3GSS)\fR
164 .ad
165 .sp .6
166 .RS 4n
167 These functions constitute the generic security services library. This library
168 is implemented as a shared object, \fBlibgss.so\fR, but it is not automatically
169 linked by the C compilation system. Specify \fB-lgss\fR on the \fBcc\fR command
170 line to link with this library. See \fBlibgss\fR(3LIB).
171 .RE

173 .sp
174 .ne 2
175 .na
176 \fB(3LDAP)\fR
177 .ad
178 .sp .6
179 .RS 4n
180 These functions constitute the lightweight directory access protocol library,
181 \fBlibldap\fR. This library is implemented as a shared object,
182 \fBlibldap.so\fR, but is not automatically linked by the C compilation system.
183 Specify \fB-lldap\fR on the \fBcc\fR command line to link with this library.
184 See \fBldap\fR(3LDAP).
185 .RE

187 .sp

```

```

188 .ne 2
189 .na
190 \fB(3NSL)\fR
191 .ad
192 .sp .6
193 .RS 4n
194 These functions constitute the network service library, \fBlibnsl\fR. This
195 library is implemented as a shared object, \fBlibnsl.so\fR, but is not
196 automatically linked by the C compilation system. Specify \fB-lnsl\fR on the
197 \fBcc\fR command line to link with this library. See \fBlibnsl\fR(3LIB).
198 .sp
199 Many base networking functions are also available in the X/Open networking
200 interfaces library, \fBlibxnet\fR. See section (3XNET) below for more
201 information on the \fBlibxnet\fR interfaces.
202 .RE

204 .sp
205 .ne 2
206 .na
207 \fB(3RESOLV)\fR
208 .ad
209 .sp .6
210 .RS 4n
211 These functions constitute the resolver library, \fBlibresolv\fR. This library
212 is implemented as a shared object, \fBlibresolv.so\fR, but is not automatically
213 linked by the C compilation system. Specify \fB-lresolv\fR on the \fBcc\fR
214 command line to link with this library. See \fBlibresolv\fR(3LIB).
215 .RE

217 .sp
218 .ne 2
219 .na
220 \fB(3RPC)\fR
221 .ad
222 .sp .6
223 .RS 4n
224 These functions constitute the remote procedure call libraries, \fBlibrpcsvc\fR
225 and \fBlibrpcsoc\fR. The latter is provided for compatibility only; new
226 applications should not link to it. Both libraries are implemented as shared
227 objects, \fBlibrpcsvc.so\fR and \fBlibrpcsoc.so\fR, respectively. Neither
228 library is automatically linked by the C compilation system. Specify
229 \fB-lrpcsvc\fR or \fB-lrpcsoc\fR on the \fBcc\fR command line to link with
230 these libraries. See \fBlibrpcsvc\fR(3LIB).
231 .RE

233 .sp
234 .ne 2
235 .na
236 \fB(3SASL)\fR
237 .ad
238 .sp .6
239 .RS 4n
240 These functions constitute the simple authentication and security layer
241 library, \fBlibsasl\fR. This library is implemented as a shared object,
242 \fBlibsasl.so\fR, but it is not automatically linked by the C compilation
243 system. Specify \fB-lsasl\fR on the \fBcc\fR command line to link with this
244 library. See \fBlibsasl\fR(3LIB).
245 .RE

247 .sp
248 .ne 2
249 .na
250 \fB(3SIP)\fR
251 .ad
252 .sp .6
253 .RS 4n

```

254 These functions constitute the session initiation protocol library,  
 255 \fBlibsip\fR. This library is implemented as a shared object, \fBlibsip.so\fR,  
 256 but it is not automatically linked by the C compilation system. Specify  
 257 \fB-lsip\fR on the \fBcc\fR command line to link with this library. See  
 258 \fBlibsip\fR(3LIB).  
 259 .RE

261 .sp  
 262 .ne 2  
 263 .na  
 264 \fB(3SLP)\fR  
 265 .ad  
 266 .sp .6  
 267 .RS 4n  
 268 These functions constitute the service location protocol library, \fBlibsip\fR.  
 269 This library is implemented as a shared object, \fBlibsip.so\fR, but it is not  
 270 automatically linked by the C compilation system. Specify \fB-lslp\fR on the  
 271 \fBcc\fR command line to link with this library. See \fBlibsip\fR(3LIB).  
 272 .RE

274 .sp  
 275 .ne 2  
 276 .na  
 277 \fB(3SOCKET)\fR  
 278 .ad  
 279 .sp .6  
 280 .RS 4n  
 281 These functions constitute the sockets library, \fBlibsocket\fR. This library  
 282 is implemented as a shared object, \fBlibsocket.so\fR, but is not automatically  
 283 linked by the C compilation system. Specify \fB-lsocket\fR on the \fBcc\fR  
 284 command line to link with this library. See \fBlibsocket\fR(3LIB).  
 285 .RE

287 .sp  
 288 .ne 2  
 289 .na  
 290 \fB(3XNET)\fR  
 291 .ad  
 292 .sp .6  
 293 .RS 4n  
 294 These functions constitute X/Open networking interfaces which comply with the  
 295 X/Open CAE Specification, Networking Services, Issue 4 (September, 1994). This  
 296 library is implemented as a shared object, \fBlibxnet.so\fR, but is not  
 297 automatically linked by the C compilation system. Specify \fB-lxnet\fR on the  
 298 \fBcc\fR command line to link with this library. See \fBlibxnet\fR(3LIB) and  
 299 \fBstandards\fR(5) for compilation information.  
 300 .RE

302 .sp  
 303 .LP  
 304 Under all circumstances, the use of the Sockets API is recommended over the XTI  
 305 and TLI APIs. If portability to other XPGV4v2 (see \fBstandards\fR(5)) systems  
 306 is a requirement, the application must use the \fBlibxnet\fR interfaces. If  
 307 portability is not required, the sockets interfaces in \fBlibsocket\fR and  
 308 \fBlibnsl\fR are recommended over those in \fBlibxnet\fR. Between the XTI and  
 309 TLI APIs, the \fBXTI\fR interfaces (available with \fBlibxnet\fR) are  
 310 recommended over the \fBXTLI\fR interfaces (available with \fBlibnsl\fR).  
 311 .SS "Curses Library Functions"  
 312 .LP  
 312 The functions described in this volume comprise the libraries that provide  
 313 graphics and character screen updating capabilities.  
 314 .sp  
 315 .ne 2  
 316 .na  
 317 \fB(3CURSES)\fR  
 318 .ad

319 .sp .6  
 320 .RS 4n  
 321 The functions constitute the following libraries:  
 322 .sp  
 323 .ne 2  
 324 .na  
 325 \fB\fBlibcurses\fR\fR  
 326 .ad  
 327 .sp .6  
 328 .RS 4n  
 329 These functions constitute the curses library, \fBlibcurses\fR. This library is  
 330 implemented as a shared object, \fBlibcurses.so\fR, but is not automatically  
 331 linked by the C compilation system. Specify \fB-lcurses\fR on the \fBcc\fR  
 332 command line to link with this library. See \fBlibcurses\fR(3LIB).  
 333 .RE

335 .sp  
 336 .ne 2  
 337 .na  
 338 \fB\fBlibform\fR\fR  
 339 .ad  
 340 .sp .6  
 341 .RS 4n  
 342 These functions constitute the forms library, \fBlibform\fR. This library is  
 343 implemented as a shared object, \fBlibform.so\fR, but is not automatically  
 344 linked by the C compilation system. Specify \fB-lform\fR on the \fBcc\fR  
 345 command line to link with this library. See \fBlibform\fR(3LIB).  
 346 .RE

348 .sp  
 349 .ne 2  
 350 .na  
 351 \fB\fBlibmenu\fR\fR  
 352 .ad  
 353 .sp .6  
 354 .RS 4n  
 355 These functions constitute the menus library, \fBlibmenu\fR. This library is  
 356 implemented as a shared object, \fBlibmenu.so\fR, but is not automatically  
 357 linked by the C compilation system. Specify \fB-lmenu\fR on the \fBcc\fR  
 358 command line to link with this library. See \fBlibmenu\fR(3LIB).  
 359 .RE

361 .sp  
 362 .ne 2  
 363 .na  
 364 \fB\fBlibpanel\fR\fR  
 365 .ad  
 366 .sp .6  
 367 .RS 4n  
 368 These functions constitute the panels library, \fBlibpanel\fR. This library is  
 369 implemented as a shared object, \fBlibpanel.so\fR, but is not automatically  
 370 linked by the C compilation system. Specify \fB-lpanel\fR on the \fBcc\fR  
 371 command line to link with this library. See \fBlibpanel\fR(3LIB).  
 372 .RE

374 .RE

376 .sp  
 377 .ne 2  
 378 .na  
 379 \fB(3XCURSES)\fR  
 380 .ad  
 381 .sp .6  
 382 .RS 4n  
 383 These functions constitute the X/Open curses library, located in  
 384 \fB/usr/xpg4/lib/libcurses.so\fR. This library provides a set of

385 internationalized functions and macros for creating and modifying input and  
 386 output to a terminal screen. Included in this library are functions for  
 387 creating windows, highlighting text, writing to the screen, reading from user  
 388 input, and moving the cursor. X/Open Curses is designed to optimize screen  
 389 update activities. The X/Open Curses library conforms fully with Issue 4 of the  
 390 X/Open Extended Curses specification. See \fBlibcurses\fR(3XCURSES).  
 391 .RE

393 .SS "Extended Library Functions"

399 .LP

394 The functions described in this volume comprise the following specialized  
 395 libraries:

396 .sp

397 .ne 2

398 .na

399 \fB(3BSM)\fR

400 .ad

401 .sp .6

402 .RS 4n

403 These functions constitute the auditing library, \fBlibbsm\fR. This  
 404 library is implemented as a shared object, \fBlibbsm.so\fR, but is not  
 405 automatically linked by the C compilation system. Specify \fB-lbsm\fR on the  
 406 \fBcc\fR command line to link with this library. See \fBlibbsm\fR(3LIB).  
 407 .RE

409 .sp

410 .ne 2

411 .na

412 \fB(3CFGADM)\fR

413 .ad

414 .sp .6

415 .RS 4n

416 These functions constitute the configuration administration library,  
 417 \fBlibcfgadm\fR. This library is implemented as a shared object,  
 418 \fBlibcfgadm.so\fR, but is not automatically linked by the C compilation  
 419 system. Specify \fB-lcfgadm\fR on the \fBcc\fR command line to link with this  
 420 library. See \fBlibcfgadm\fR(3LIB).  
 421 .RE

423 .sp

424 .ne 2

425 .na

426 \fB(3CONTRACT)\fR

427 .ad

428 .sp .6

429 .RS 4n

430 These functions constitute the contract management library, \fBlibcontract\fR.  
 431 This library is implemented as a shared object, \fBlibcontract.so\fR, but is  
 432 not automatically linked by the C compilation system. Specify \fB-lcontract\fR  
 433 on the \fBcc\fR command line to link with this library. See  
 434 \fBlibcontract\fR(3LIB).  
 435 .RE

437 .sp

438 .ne 2

439 .na

440 \fB(3CPC)\fR

441 .ad

442 .sp .6

443 .RS 4n

444 These functions constitute the CPU performance counter library, \fBlibpcp\fR,  
 445 and the process context library, \fBlibpctx\fR. These libraries are implemented  
 446 as shared objects, \fBlibpcp.so\fR and \fBlibpctx.so\fR, respectively, but are  
 447 not automatically linked by the C compilation system. Specify \fB-lpcp\fR or  
 448 \fB-lpctx\fR on the \fBcc\fR command line to link with these libraries. See  
 449 \fBlibpcp\fR(3LIB) and \fBlibpctx\fR(3LIB).

450 .RE

452 .sp

453 .ne 2

454 .na

455 \fB(3DAT)\fR

456 .ad

457 .sp .6

458 .RS 4n

459 These functions constitute the direct access transport library, \fBlibdat\fR.  
 460 This library is implemented as a shared object, \fBlibdat.so\fR, but is not  
 461 automatically linked by the C compilation system. Specify \fB-ladat\fR on the  
 462 \fBcc\fR command line to link with this library. See \fBlibdat\fR(3LIB).  
 463 .RE

465 .sp

466 .ne 2

467 .na

468 \fB(3DEVID)\fR

469 .ad

470 .sp .6

471 .RS 4n

472 These functions constitute the device \fBID\fR library, \fBlibdevid\fR. This  
 473 library is implemented as a shared object, \fBlibdevid.so\fR, but is not  
 474 automatically linked by the C compilation system. Specify \fB-ldevid\fR on the  
 475 \fBcc\fR command line to link with this library. See \fBlibdevid\fR(3LIB).  
 476 .RE

478 .sp

479 .ne 2

480 .na

481 \fB(3DEVINFO)\fR

482 .ad

483 .sp .6

484 .RS 4n

485 These functions constitute the device information library, \fBlibdevinfo\fR.  
 486 This library is implemented as a shared object, \fBlibdevinfo.so\fR, but is not  
 487 automatically linked by the C compilation system. Specify \fB-ldevinfo\fR on  
 488 the \fBcc\fR command line to link with this library. See  
 489 \fBlibdevinfo\fR(3LIB).  
 490 .RE

492 .sp

493 .ne 2

494 .na

495 \fB(3ELF)\fR

496 .ad

497 .sp .6

498 .RS 4n

499 These functions constitute the ELF access library, \fBlibelf\fR, (Extensible  
 500 Linking Format). This library provides the interface for the creation and  
 501 analyses of "elf" files; executables, objects, and shared objects. \fBlibelf\fR  
 502 is implemented as a shared object, \fBlibelf.so\fR, but is not automatically  
 503 linked by the C compilation system. Specify \fB-lelf\fR on the \fBcc\fR command  
 504 line to link with this library. See \fBlibelf\fR(3LIB).  
 505 .RE

507 .sp

508 .ne 2

509 .na

510 \fB(3EXACCT)\fR

511 .ad

512 .sp .6

513 .RS 4n

514 These functions constitute the extended accounting access library,  
 515 \fBlibexact\fR, and the project database access library, \fBlibproject\fR.

516 These libraries are implemented as shared objects, \fBlibexact.so\fR and  
 517 \fBlibproject.so\fR, respectively, but are not automatically linked by the C  
 518 compilation system. Specify \fBlexact\fR or \fBlproject\fR on the \fBcc\fR  
 519 command line to link with these libraries. See \fBlibexact\fR(3LIB) and  
 520 \fBlibproject\fR(3LIB).  
 521 .RE

523 .sp  
 524 .ne 2  
 525 .na  
 526 \fB(3FCOE)\fR  
 527 .ad  
 528 .sp .6  
 529 .RS 4n  
 530 These functions constitute the Fibre Channel over Ethernet port management  
 531 library. This library is implemented as a shared object, \fBlibfcoe.so\fR, but  
 532 is not automatically linked by the C compilation system. Specify \fBlfcoe\fR  
 533 on the \fBcc\fR command line to link with this library. See  
 534 \fBlibfcoe\fR(3LIB).  
 535 .RE

537 .sp  
 538 .ne 2  
 539 .na  
 540 \fB(3FSTYP)\fR  
 541 .ad  
 542 .sp .6  
 543 .RS 4n  
 544 These functions constitute the file system type identification library. This  
 545 library is implemented as a shared object, \fBlibfstyp.so\fR, but is not  
 546 automatically linked by the C compilation system. Specify \fBlfstyp\fR on the  
 547 \fBcc\fR command line to link with this library. See \fBlibfstyp\fR(3LIB).  
 548 .RE

550 .sp  
 551 .ne 2  
 552 .na  
 553 \fB(3GEN)\fR  
 554 .ad  
 555 .sp .6  
 556 .RS 4n  
 557 These functions constitute the string pattern-matching and pathname  
 558 manipulation library, \fBlibgen\fR. This library is implemented as a shared  
 559 object, \fBlibgen.so\fR, but is not automatically linked by the C compilation  
 560 system. Specify \fBlgen\fR on the \fBcc\fR command line to link with this  
 561 library. See \fBlibgen\fR(3LIB).  
 562 .RE

564 .sp  
 565 .ne 2  
 566 .na  
 567 \fB(3HBAAPI)\fR  
 568 .ad  
 569 .sp .6  
 570 .RS 4n  
 571 These functions constitute the common fibre channel HBA information library,  
 572 \fBlibhbaapi\fR. This library is implemented as a shared object,  
 573 \fBlibhbaapi.so\fR, but is not automatically linked by the C compilation  
 574 system. Specify \fBlhbaapi\fR on the \fBcc\fR command line to link with this  
 575 library. See \fBlibhbaapi\fR(3LIB).  
 576 .RE

578 .sp  
 579 .ne 2  
 580 .na  
 581 \fB(3ISCSIT)\fR

582 .ad  
 583 .sp .6  
 584 .RS 4n  
 585 These functions constitute the iSCSI Management library, \fBlibiscsit\fR. This  
 586 library is implemented as a shared object, \fBlibiscsit.so\fR, but is not  
 587 automatically linked by the C compilation system. Specify \fBliscsit\fR on the  
 588 \fBcc\fR command line to link with this library. See \fBlibiscsit\fR(3LIB).  
 589 .RE

591 .sp  
 592 .ne 2  
 593 .na  
 594 \fB(3KSTAT)\fR  
 595 .ad  
 596 .sp .6  
 597 .RS 4n  
 598 These functions constitute the kernel statistics library, which is implemented  
 599 as a shared object, \fBlibkstat.so\fR, but is not automatically linked by the C  
 600 compilation system. Specify \fBlkstat\fR on the \fBcc\fR command line to link  
 601 with this library. See \fBlibkstat\fR(3LIB).  
 602 .RE

604 .sp  
 605 .ne 2  
 606 .na  
 607 \fB(3KVM)\fR  
 608 .ad  
 609 .sp .6  
 610 .RS 4n  
 611 These functions allow access to the kernel's virtual memory library, which is  
 612 implemented as a shared object, \fBlibkvm.so\fR, but is not automatically  
 613 linked by the C compilation system. Specify \fBlkvm\fR on the \fBcc\fR command  
 614 line to link with this library. See \fBlibkvm\fR(3LIB).  
 615 .RE

617 .sp  
 618 .ne 2  
 619 .na  
 620 \fB(3LGRP)\fR  
 621 .ad  
 622 .sp .6  
 623 .RS 4n  
 624 These functions constitute the locality group library, which is implemented as  
 625 a shared object, \fBliblgrp.so\fR, but is not automatically linked by the C  
 626 compilation system. Specify \fBllgrp\fR on the \fBcc\fR command line to link  
 627 with this library. See \fBliblgrp\fR(3LIB).  
 628 .RE

630 .sp  
 631 .ne 2  
 632 .na  
 633 \fB(3M)\fR  
 634 .ad  
 635 .sp .6  
 636 .RS 4n  
 637 These functions constitute the mathematical library, \fBlibm\fR. This library  
 638 is implemented as a shared object, \fBlibm.so\fR, but is not automatically  
 639 linked by the C compilation system. Specify \fBlm\fR on the \fBcc\fR command  
 640 line to link with this library. See \fBlibm\fR(3LIB).  
 641 .RE

643 .sp  
 644 .ne 2  
 645 .na  
 646 \fB(3MAIL)\fR  
 647 .ad

```

648 .sp .6
649 .RS 4n
650 These functions constitute the user mailbox management library, \fBlibmail\fR.
651 This library is implemented as a shared object, \fBlibmail.so\fR, but is not
652 automatically linked by the C compilation system. Specify \fB-lmail\fR on the
653 \fBcc\fR command line to link with this library. See \fBlibmail\fR(3LIB).
654 .RE

656 .sp
657 .ne 2
658 .na
659 \fB(3MP)\fR
660 .ad
661 .sp .6
662 .RS 4n
663 These functions constitute the integer mathematical library, \fBlibmp\fR. This
664 library is implemented as a shared object, \fBlibmp.so\fR, but is not
665 automatically linked by the C compilation system. Specify \fB-lmp\fR on the
666 \fBcc\fR command line to link with this library. See \fBlibmp\fR(3LIB).
667 .RE

669 .sp
670 .ne 2
671 .na
672 \fB(3MPAPI)\fR
673 .ad
674 .sp .6
675 .RS 4n
676 These functions constitute the Common Multipath Management library,
677 \fBlibMPAPI\fR. This library is implemented as a shared object,
678 \fBlibMPAPI.so\fR, but is not automatically linked by the C compilation system.
679 Specify \fB-lMPAPI\fR on the \fBcc\fR command line to link with this library.
680 See \fBlibMPAPI\fR(3LIB).
681 .RE

683 .sp
684 .ne 2
685 .na
686 \fB(3MVEC)\fR
687 .ad
688 .sp .6
689 .RS 4n
690 These functions constitute the vector mathematical library, \fBlibmvec\fR. This
691 library is implemented as a shared object, \fBlibmvec.so\fR, but is not
692 automatically linked by the C compilation system. Specify \fB-lmvec\fR on the
693 \fBcc\fR command line to link with this library. See \fBlibmvec\fR(3LIB).
694 .RE

696 .sp
697 .ne 2
698 .na
699 \fB(3NVPAR)\fR
700 .ad
701 .sp .6
702 .RS 4n
703 These functions constitute the name-value pair library, \fBlibnvpair\fR. This
704 library is implemented as a shared object, \fBlibnvpair.so\fR, but is not
705 automatically linked by the C compilation system. Specify \fB-lnvpair\fR on the
706 \fBcc\fR command line to link with this library. See \fBlibnvpair\fR(3LIB).
707 .RE

709 .sp
710 .ne 2
711 .na
712 \fB(3PAM)\fR
713 .ad

```

```

714 .sp .6
715 .RS 4n
716 These functions constitute the pluggable authentication module library,
717 \fBlibpam\fR. This library is implemented as a shared object, \fBlibpam.so\fR,
718 but is not automatically linked by the C compilation system. Specify
719 \fB-lpam\fR on the \fBcc\fR command line to link with this library. See
720 \fBlibpam\fR(3LIB).
721 .RE

723 .sp
724 .ne 2
725 .na
726 \fB(3PAPI)\fR
727 .ad
728 .sp .6
729 .RS 4n
730 These functions constitute the Free Standards Group Open Printing API (PAPI)
731 library, \fBlibpapi\fR. This library is implemented as a shared object,
732 \fBlibpapi.so\fR, but is not automatically linked by the C compilation system.
733 Specify \fB-lpapi\fR on the \fBcc\fR command line to link with this library.
734 See \fBlibpapi\fR(3LIB).
735 .RE

737 .sp
738 .ne 2
739 .na
740 \fB(3PICL)\fR
741 .ad
742 .sp .6
743 .RS 4n
744 These functions constitute the PICL library, \fBlibpicl\fR. This library is
745 implemented as a shared object, \fBlibpicl.so\fR, but is not automatically
746 linked by the C compilation system. Specify \fB-lpicl\fR on the \fBcc\fR
747 command line to link with this library. See \fBlibpicl\fR(3LIB) and
748 \fBlibpicl\fR(3PICL).
749 .RE

751 .sp
752 .ne 2
753 .na
754 \fB(3PICLTREE)\fR
755 .ad
756 .sp .6
757 .RS 4n
758 These functions constitute the PICL plug-in library, \fBlibpicltree\fR. This
759 library is implemented as a shared object, \fBlibpicltree.so\fR, but is not
760 automatically linked by the C compilation system. Specify \fB-lpicltree\fR on
761 the \fBcc\fR command line to link with this library. See
762 \fBlibpicltree\fR(3LIB) and \fBlibpicltree\fR(3PICLTREE).
763 .RE

765 .sp
766 .ne 2
767 .na
768 \fB(3POOL)\fR
769 .ad
770 .sp .6
771 .RS 4n
772 These functions constitute the pool configuration manipulation library,
773 \fBlibpool\fR. This library is implemented as a shared object,
774 \fBlibpool.so\fR, but is not automatically linked by the C compilation system.
775 Specify \fB-lpool\fR on the \fBcc\fR command line to link with this library.
776 See \fBlibpool\fR(3LIB).
777 .RE

779 .sp

```

```

780 .ne 2
781 .na
782 \fB(3PROC)\fR
783 .ad
784 .sp .6
785 .RS 4n
786 These functions constitute the process manipulation library,
787 \fBlibproc\fR. This library is implemented as a shared object,
788 \fBlibproc.so\fR, but it is not automatically linked by the C compilation
789 system. Specify \fB-lproc\fR on the \fBcc\fR command line to link with this
790 library. See \fBlibproc\fR(3LIB).
791 .RE

793 .sp
794 .ne 2
795 .na
796 \fB(3PROJECT)\fR
797 .ad
798 .sp .6
799 .RS 4n
800 These functions constitute the project database access library,
801 \fBlibproject\fR. This library is implemented as a shared object,
802 \fBlibproject.so\fR, but is not automatically linked by the C compilation
803 system. Specify \fB-lproject\fR on the \fBcc\fR command line to link with this
804 library. See \fBlibproject\fR(3LIB).
805 .RE

807 .sp
808 .ne 2
809 .na
810 \fB(3RSM)\fR
811 .ad
812 .sp .6
813 .RS 4n
814 These functions constitute the remote shared memory library, \fBlibrsm\fR. This
815 library is implemented as a shared object, \fBlibrsm.so\fR, but is not
816 automatically linked by the C compilation system. Specify \fB-lrsm\fR on the
817 \fBcc\fR command line to link with this library. See \fBlibrsm\fR(3LIB).
818 .RE

820 .sp
821 .ne 2
822 .na
823 \fB(3SCF)\fR
824 .ad
825 .sp .6
826 .RS 4n
827 These functions constitute the service configuration facility library,
828 These functions constitute the object-caching memory allocation library,
829 \fBlibscf\fR. This library is implemented as a shared object, \fBlibscf.so\fR,
830 but is not automatically linked by the C compilation system. Specify
831 \fB-lscf\fR on the \fBcc\fR command line to link with this library. See
832 \fBlibscf\fR(3LIB).
833 .RE

834 .sp
835 .ne 2
836 .na
837 \fB(3SEC)\fR
838 .ad
839 .sp .6
840 .RS 4n
841 These functions constitute the file access control library, \fBlibsec\fR. This
842 library is implemented as a shared object, \fBlibsec.so\fR, but is not
843 automatically linked by the C compilation system. Specify \fB-lsec\fR on the
844 \fBcc\fR command line to link with this library. See \fBlibsec\fR(3LIB).

```

```

845 .RE

847 .sp
848 .ne 2
849 .na
850 \fB(3SECDB)\fR
851 .ad
852 .sp .6
853 .RS 4n
854 These functions constitute the security attributes database library,
855 \fBlibsecdb\fR. This library is implemented as a shared object,
856 \fBlibsecdb.so\fR, but is not automatically linked by the C compilation system.
857 Specify \fB-lsecdb\fR on the \fBcc\fR command line to link with this library.
858 See \fBlibsecdb\fR(3LIB).
859 .RE

861 .sp
862 .ne 2
863 .na
864 \fB(3STMF)\fR
865 .ad
866 .sp .6
867 .RS 4n
868 These functions constitute the SCSI Target Mode Framework library,
869 \fBlibstmf\fR. This library is implemented as a shared object,
870 \fBlibstmf.so\fR, but is not automatically linked by the C compilation system.
871 Specify \fB-lstmf\fR on the \fBcc\fR command line to link with this library.
872 See \fBlibstmf\fR(3LIB).
873 .RE

875 .sp
876 .ne 2
877 .na
878 \fB(3SYSEVENT)\fR
879 .ad
880 .sp .6
881 .RS 4n
882 These functions constitute the system event library, \fBlibsysevent\fR. This
883 library is implemented as a shared object, \fBlibsysevent.so\fR, but is not
884 automatically linked by the C compilation system. Specify \fB-lsysevent\fR on
885 the \fBcc\fR command line to link with this library. See
886 \fBlibsysevent\fR(3LIB).
887 .RE

889 .sp
890 .ne 2
891 .na
892 \fB(3TECLA)\fR
893 .ad
894 .sp .6
895 .RS 4n
896 These functions constitute the interactive command-line input library,
897 \fBlibtecla\fR. This library is implemented as a shared object,
898 \fBlibtecla.so\fR, but is not automatically linked by the C compilation system.
899 Specify \fB-ltecla\fR on the \fBcc\fR command line to link with this library.
900 See \fBlibtecla\fR(3LIB).
901 .RE

903 .sp
904 .ne 2
905 .na
906 \fB(3TNF)\fR
907 .ad
908 .sp .6
909 .RS 4n
910 These functions constitute the TNF libraries, \fBlibtnf\fR, \fBlibtnfctl\fR,

```

911 and `\fBlibtnfprobe`. These libraries are implemented as shared objects,  
 912 `\fBlibtnf.so`, `\fBlibtnfctl.so`, and `\fBlibtnfprobe.so`, respectively,  
 913 but are not automatically linked by the C compilation system. Specify  
 914 `\fB-ltnf`, `\fB-ltnfctl`, or `\fB-ltnfprobe` on the `\fBcc` command line  
 915 to link with these libraries. See `\fBlibtnfctl`(3TNF) and  
 916 `\fBlibtnfctl`(3LIB).  
 917 .RE

919 .sp  
 920 .ne 2  
 921 .na  
 922 `\fB(3TSOL)`  
 923 .ad  
 924 .sp .6  
 925 .RS 4n  
 926 These functions constitute the Trusted Extensions library, `\fBlibtsol`, and  
 927 the Trusted Extensions network library, `\fBlibtsnet`. These libraries are  
 928 implemented as shared objects, `\fBlibtsol.so` and `\fBlibtsnet.so`, but are  
 929 not automatically linked by the C compilation system. Specify `\fB-ltsol` or  
 930 `\fB-ltsnet` on the `\fBcc` command line to link with these libraries. See  
 931 `\fBlibtsol`(3LIB) and `\fBlibtsnet`(3LIB).  
 932 .RE

934 .sp  
 935 .ne 2  
 936 .na  
 937 `\fB(3UUIID)`  
 938 .ad  
 939 .sp .6  
 940 .RS 4n  
 941 These functions constitute the universally unique identifier library,  
 942 `\fBlibuuiid`. This library is implemented as a shared object,  
 943 `\fBlibuuiid.so`, but is not automatically linked by the C compilation system.  
 944 Specify `\fB-luuiid` on the `\fBcc` command line to link with this library.  
 945 See `\fBlibuuiid`(3LIB).  
 946 .RE

948 .sp  
 949 .ne 2  
 950 .na  
 951 `\fB(3VOLMGT)`  
 952 .ad  
 953 .sp .6  
 954 .RS 4n  
 955 These functions constitute the volume management library, `\fBlibvolmgt`. This  
 956 library is implemented as a shared object, `\fBlibvolmgt.so`, but is not  
 957 automatically linked by the C compilation system. Specify `\fB-lvolmgt` on the  
 958 `\fBcc` command line to link with this library. See `\fBlibvolmgt`(3LIB).  
 959 .RE

961 .SH DEFINITIONS  
 962 .LP  
 962 A character is any bit pattern able to fit into a byte on the machine. In some  
 963 international languages, however, a "character" might require more than one  
 964 byte, and is represented in multi-bytes.  
 965 .sp  
 966 .LP  
 967 The null character is a character with value 0, conventionally represented in  
 968 the C language as `\fB\e|0`. A character array is a sequence of characters.  
 969 A null-terminated character array (a `\fBstring`) is a sequence of characters,  
 970 the last of which is the null character. The null string is a character array  
 971 containing only the terminating null character. A null pointer is the value  
 972 that is obtained by casting `\fB0` into a pointer. C guarantees that this  
 973 value will not match that of any legitimate pointer, so many functions that  
 974 return pointers return `\fBNULL` to indicate an error. The macro `\fBNULL` is  
 975 defined in `<\fBstdio.h`. Types of the form `\fBsize_t` are defined in the

976 appropriate headers.  
 977 .SH MULTITHREADED APPLICATIONS  
 985 .LP  
 978 Both POSIX threads and Solaris threads can be used within the same application.  
 979 Their implementations are completely compatible with each other; however, only  
 980 POSIX threads guarantee portability to other POSIX-conforming environments.  
 981 .sp  
 982 .LP  
 983 The `\fBlibpthread` and `\fBlibthread` libraries are implemented  
 984 as filters on `\fBlibc`.  
 985 .sp  
 986 .LP  
 987 When compiling a multithreaded application, the `\fB-mt` option must be  
 988 specified on the command line.  
 989 .sp  
 990 .LP  
 991 There is no need for a multithreaded application to link with `\fB-lthread`.  
 992 An application must link with `\fB-lpthread` only when POSIX semantics for  
 993 `\fBfork` are desired. When an application is linked with `\fB-lpthread`,  
 994 a call to `\fBfork` assumes the behavior `\fBfork1` rather than the  
 995 default behavior that forks all threads.  
 996 .sp  
 997 .LP  
 998 When compiling a POSIX-conforming application, either the `\fB_POSIX_C_SOURCE`  
 999 or `\fB_POSIX_PTHREAD_SEMANTICS` option must be specified on the command line.  
 1000 For POSIX.1c-conforming applications, define the `\fB_POSIX_C_SOURCE` flag to  
 1001 be `>= 199506L`:  
 1002 .sp  
 1003 .in +2  
 1004 .nf  
 1005 `\fBcc \fB-mt [ \fBiflag \fB... ] \fBifile \fB... \fB-D_POSIX_C_SOURCE=199506L \fB`  
 1006 .fi  
 1007 .in -2

1009 .sp  
 1010 .LP  
 1011 For POSIX behavior with the Solaris `\fBfork` and `\fBfork1` distinction,  
 1012 compile as follows:  
 1013 .sp  
 1014 .in +2  
 1015 .nf  
 1016 `\fBcc \fB-mt [ \fBiflag \fB... ] \fBifile \fB... \fB-D_POSIX_PTHREAD_SEMANTICS \fB`  
 1017 .fi  
 1018 .in -2

1020 .sp  
 1021 .LP  
 1022 For Solaris threads behavior, compile as follows:  
 1023 .sp  
 1024 .in +2  
 1025 .nf  
 1026 `\fBcc \fB-mt [ \fBiflag \fB... ] \fBifile \fB... \fB`  
 1027 .fi  
 1028 .in -2

1030 .sp  
 1031 .LP  
 1032 Unsafe interfaces should be called only from the main thread to ensure the  
 1033 application's safety.  
 1034 .sp  
 1035 .LP  
 1036 MT-Safe interfaces are denoted in the `\fBATTRIBUTES` section of the functions  
 1037 and libraries manual pages (see `\fBattributes`(5)). If a manual page does not  
 1038 state explicitly that an interface is MT-Safe, the user should assume that the  
 1039 interface is unsafe.  
 1040 .SH REALTIME APPLICATIONS



```

1049 .LP
1041 The environment variable \fBLD_BIND_NOW must be set to a non-null value to
1042 enable early binding. Refer to the "When Relocations are Processed" chapter in
1043 \fILinker and Libraries Guide for additional information.
1044 .SH FILES
1045 .ne 2
1046 .na
1047 \fB\fIINCDIR
1048 .ad
1049 .RS 15n
1050 usually \fB/usr/include
1051 .RE

1053 .sp
1054 .ne 2
1055 .na
1056 \fB\fILIBDIR
1057 .ad
1058 .RS 15n
1059 usually either \fB/lib or \fB/usr/lib (32-bit) or either \fB/lib/64 or
1060 \fB/usr/lib/64 (64-bit)
1061 .RE

1063 .sp
1064 .ne 2
1065 .na
1066 \fB\fILIBDIR \fB/*.so
1067 .ad
1068 .RS 15n
1069 shared libraries
1070 .RE

1072 .SH ACKNOWLEDGMENTS
1082 .LP
1073 Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission to
1074 reproduce portions of its copyrighted documentation. Original documentation
1075 from The Open Group can be obtained online at
1076 http://www.opengroup.org/bookstore/.
1077 .sp
1078 .LP
1079 The Institute of Electrical and Electronics Engineers and The Open Group, have
1080 given us permission to reprint portions of their documentation.
1081 .sp
1082 .LP
1083 In the following statement, the phrase "this text" refers to portions of the
1084 system documentation.
1085 .sp
1086 .LP
1087 Portions of this text are reprinted and reproduced in electronic form in the
1088 SunOS Reference Manual, from IEEE Std 1003.1, 2004 Edition, Standard for
1089 Information Technology -- Portable Operating System Interface (POSIX), The Open
1090 Group Base Specifications Issue 6, Copyright (C) 2001-2004 by the Institute of
1091 Electrical and Electronics Engineers, Inc and The Open Group. In the event of
1092 any discrepancy between these versions and the original IEEE and The Open Group
1093 Standard, the original IEEE and The Open Group Standard is the referee
1094 document. The original Standard can be obtained online at
1095 http://www.opengroup.org/unix/online.html.
1096 .sp
1097 .LP
1098 This notice shall appear on any product containing this material.
1099 .SH SEE ALSO
1110 .LP
1100 \fBar(1), \fBld(1), \fBfork(2), \fBstdio(3C), \fBattributes(5),
1101 \fBstandards(5)
1102 .sp
1103 .LP

```

```

1104 \fILinker and Libraries Guide
1105 .sp
1106 .LP
1107 \fIPerformance Profiling Tools
1108 .sp
1109 .LP
1110 \fIANSI C Programmer's Guide
1111 .SH DIAGNOSTICS
1123 .LP
1112 For functions that return floating-point values, error handling varies
1113 according to compilation mode. Under the \fB-Xt (default) option to
1114 \fBcc, these functions return the conventional values \fB0,
1115 \fB(+HUGE), or \fBNaN when the function is undefined for the given
1116 arguments or when the value is not representable. In the \fB-Xa and
1117 \fB-Xc compilation modes, \fB(+HUGE_VAL) is returned instead of
1118 \fB(+HUGE). (\fBHUGE_VAL and \fBHUGE are defined in \fBmath.h to
1119 be infinity and the largest-magnitude single-precision number, respectively.)
1120 .SH NOTES
1133 .LP
1121 None of the functions, external variables, or macros should be redefined in the
1122 user's programs. Any other name can be redefined without affecting the behavior
1123 of other library functions, but such redefinition might conflict with a
1124 declaration in an included header.
1125 .sp
1126 .LP
1127 The headers in \fIINCDIR provide function prototypes (function declarations
1128 including the types of arguments) for most of the functions listed in this
1129 manual. Function prototypes allow the compiler to check for correct usage of
1130 these functions in the user's program. The \fBlint program checker can also
1131 be used and will report discrepancies even if the headers are not included with
1132 \fB#include statements. Definitions for Sections 2 and 3C are checked
1133 automatically. Other definitions can be included by using the \fB-l option
1134 to \fBlint. (For example, \fB-lm includes definitions for \fBlibm.)
1135 Use of \fBlint is highly recommended. See the \fBlint chapter in
1136 \fIPerformance Profiling Tools
1137 .sp
1138 .LP
1139 Users should carefully note the difference between STREAMS and \fIstream.
1140 STREAMS is a set of kernel mechanisms that support the development of network
1141 services and data communication drivers. It is composed of utility routines,
1142 kernel facilities, and a set of data structures. A \fIstream is a file with
1143 its associated buffering. It is declared to be a pointer to a type \fBFILE
1144 defined in \fB<stdio.h>.
1145 .sp
1146 .LP
1147 In detailed definitions of components, it is sometimes necessary to refer to
1148 symbolic names that are implementation-specific, but which are not necessarily
1149 expected to be accessible to an application program. Many of these symbolic
1150 names describe boundary conditions and system limits.
1151 .sp
1152 .LP
1153 In this section, for readability, these implementation-specific values are
1154 given symbolic names. These names always appear enclosed in curly brackets to
1155 distinguish them from symbolic names of other implementation-specific constants
1156 that are accessible to application programs by headers. These names are not
1157 necessarily accessible to an application program through a header, although
1158 they can be defined in the documentation for a particular system.
1159 .sp
1160 .LP
1161 In general, a portable application program should not refer to these symbolic
1162 names in its code. For example, an application program would not be expected to
1163 test the length of an argument list given to a routine to determine if it was
1164 greater than {\fBARG_MAX}.

```