```
**********************************************************
    2969 Mon Dec 28 20:01:49 2015
new/usr/src/head/priv.h
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
```

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License (the "License").
 * You may not use this file except in compliance with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright 2014 Garrett D'Amore <garrett@damore.org>
 *
 * Copyright 2010 Sun Microsystems, Inc.  All rights reserved.
 * Use is subject to license terms.
 */

#ifndef _PRIV_H_
#define _PRIV_H_

#include <sys/priv.h>

#ifdef  __cplusplus
extern "C" {
#endif

#define PRIV_STR_PORT           0x00                    /* portable output */
#define PRIV_STR_LIT            0x01                    /* literal output */
#define PRIV_STR_SHORT          0x02                    /* shortest output */

#define PRIV_ALLSETS            ((priv_ptype_t)0)       /* for priv_set() */

/*
 * library functions prototype.
 */

extern int setppriv(priv_op_t, priv_ptype_t, const priv_set_t *);
extern int getppriv(priv_ptype_t, priv_set_t *);
extern int setpflags(uint_t, uint_t);
extern uint_t getpflags(uint_t);
extern const priv_impl_info_t *getprivimplinfo(void);

extern int priv_set(priv_op_t, priv_ptype_t, ...);
extern boolean_t priv_ineffect(const char *);
extern priv_set_t *priv_str_to_set(const char *, const char *, const char **);
extern char *priv_set_to_str(const priv_set_t *, char, int);

extern int priv_getbyname(const char *);
extern const char *priv_getbynum(int);
extern int priv_getsetbyname(const char *);
extern const char *priv_getsetbynum(int);
```



```
extern char *priv_gettext(const char *);

extern priv_set_t *priv_allocset(void);
extern void priv_freeset(priv_set_t *);

extern void priv_emptyset(priv_set_t *);
extern void priv_basicset(priv_set_t *);
extern void priv_defaultset(priv_set_t *);
#endif /* ! codereview */
extern void priv_fillset(priv_set_t *);
extern boolean_t priv_isemptyset(const priv_set_t *);
extern boolean_t priv_isfullset(const priv_set_t *);
extern boolean_t priv_isequalset(const priv_set_t *, const priv_set_t *);
extern boolean_t priv_issubset(const priv_set_t *, const priv_set_t *);
extern void priv_intersect(const priv_set_t *, priv_set_t *);
extern void priv_union(const priv_set_t *, priv_set_t *);
extern void priv_inverse(priv_set_t *);
extern int priv_addset(priv_set_t *, const char *);
extern void priv_copyset(const priv_set_t *, priv_set_t *);
extern int priv_delset(priv_set_t *, const char *);
extern boolean_t priv_ismember(const priv_set_t *, const char *);

#ifdef  __cplusplus
}
#endif

#endif  /* _PRIV_H_ */
```

```
**********************************************************
    3544 Mon Dec 28 20:01:53 2015
new/usr/src/lib/libc/inc/priv_private.h
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
```

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License (the "License").
 * You may not use this file except in compliance with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */

/*
 * Copyright 2007 Sun Microsystems, Inc.  All rights reserved.
 * Use is subject to license terms.
 */

#ifndef _PRIV_PRIVATE_H
#define _PRIV_PRIVATE_H

#pragma ident   "%Z%%M% %I%     %E% SMI"

#include <sys/types.h>
#include <sys/priv.h>
#include <limits.h>

/*
 * Libc private privilege data.
 */

#ifdef __cplusplus
extern "C" {
#endif

#define LOADPRIVDATA(d)         d = __priv_getdata()
#define GETPRIVDATA()           __priv_getdata()
#define LOCKPRIVDATA()          { \
                                        /* Data already allocated */ \
                                        (void) lock_data(); \
                                        (void) refresh_data(); \
                                }
#define UNLOCKPRIVDATA()        unlock_data()
#define WITHPRIVLOCKED(t, b, x) { \
                                        t __result; \
                                        if (lock_data() != 0) \
                                                return (b); \
                                        __result = (x); \
                                        if (__result == (b) && refresh_data()) \
                                                __result = (x); \
                                        unlock_data(); \
                                        return (__result); \
                                }
```



```
/*
 * Privilege mask macros.
 */
#define __NBWRD         (CHAR_BIT * sizeof (priv_chunk_t))
#define privmask(n)     (1 << ((__NBWRD - 1) - ((n) % __NBWRD)))
#define privword(n)     ((n)/__NBWRD)

/*
 * Same as the functions, but for numeric privileges.
 */
#define PRIV_ADDSET(a, p)       ((priv_chunk_t *)(a))[privword(p)] |= \
                                                        privmask(p)
#define PRIV_DELSET(a, p)       ((priv_chunk_t *)(a))[privword(p)] &= \
                                                        ~privmask(p)
#define PRIV_ISMEMBER(a, p)     ((((priv_chunk_t *)(a))[privword(p)] & \
                                                        privmask(p)) != 0)

/*
 * The structure is static except for the setsort, privnames and nprivs
 * field.  The pinfo structure initially has sufficient room and the kernel
 * guarantees no offset changes so we can copy a new structure on top of it.
 * The locking stratgegy is this: we lock it when we need to reference any
 * of the volatile fields.
 */
typedef struct priv_data {
        size_t                  pd_setsize;             /* In bytes */
        int                     pd_nsets, pd_nprivs;
        uint32_t                pd_ucredsize;
        char                    **pd_setnames;
        char                    **pd_privnames;
        int                     *pd_setsort;
        priv_impl_info_t        *pd_pinfo;
        priv_set_t              *pd_basicset;
        priv_set_t              *pd_defaultset;
#endif /* ! codereview */
        priv_set_t              *pd_zoneset;
} priv_data_t;

extern priv_data_t *__priv_getdata(void);
extern priv_data_t *__priv_parse_info(priv_impl_info_t *);
extern void __priv_free_info(priv_data_t *);
extern priv_data_t *privdata;

extern int lock_data(void);
extern boolean_t refresh_data(void);
extern void unlock_data(void);

extern boolean_t __priv_isemptyset(priv_data_t *, const priv_set_t *);
extern boolean_t __priv_isfullset(priv_data_t *, const priv_set_t *);
extern boolean_t __priv_issubset(priv_data_t *, const priv_set_t *,
                                 const priv_set_t *);
extern const char *__priv_getbynum(const priv_data_t *, int);

extern int getprivinfo(priv_impl_info_t *, size_t);

extern priv_set_t *priv_basic(void);
extern priv_set_t *priv_default(void);
#endif /* ! codereview */

#ifdef __cplusplus
}
#endif

#endif /* _PRIV_PRIVATE_H */
```

```
**********************************************************
   11111 Mon Dec 28 20:01:58 2015
new/usr/src/lib/libc/port/gen/priv_str_xlate.c
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
______unchanged_portion_omitted_
```

```
 61 priv_set_t *
 62 priv_default(void)
 63 {
 64         priv_data_t *d;

 66         LOADPRIVDATA(d);

 68         return (d->pd_defaultset);
 69 }

 71 #endif /* ! codereview */
 72 /*
 73  *      Name:   priv_str_to_set()
 74  *
 75  *      Description:    Given a buffer with privilege strings, the
 76  *      equivalent privilege set is returned.
 77  *
 78  *      Special tokens recognized: all, none, basic and "".
 79  *
 80  *      On failure, this function returns NULL.
 81  *      *endptr == NULL and errno set: resource error.
 82  *      *endptr != NULL: parse error.
 83  */
 84 priv_set_t *
 85 priv_str_to_set(const char *priv_names,
 86                 const char *separators,
 87                 const char **endptr)
 88 {

 90         char *base;
 91         char *offset;
 92         char *last;
 93         priv_set_t *pset = NULL;
 94         priv_set_t *zone = NULL;
 95         priv_set_t *basic = NULL;
 96         priv_set_t *deflt = NULL;
 61         priv_set_t *zone;
 62         priv_set_t *basic;

 98         if (endptr != NULL)
 99                 *endptr = NULL;

101         if ((base = libc_strdup(priv_names)) == NULL ||
102             (pset = priv_allocset()) == NULL) {
103                 /* Whether base is NULL or allocated, this works */
104                 libc_free(base);
105                 return (NULL);
106         }

108         priv_emptyset(pset);
109         basic = priv_basic();
110         deflt = priv_default();
111 #endif /* ! codereview */
112         zone = privdata->pd_zoneset;

114         /* This is how to use strtok_r nicely in a while loop ... */
115         last = base;

117         while ((offset = strtok_r(NULL, separators, &last)) != NULL) {
```



```
118                 /*
119                  * Search for these special case strings.
120                  */
121                 if (basic != NULL && strcasecmp(offset, "basic") == 0) {
122                         priv_union(basic, pset);
123                 } else if (deflt != NULL && strcasecmp(offset,
124                     "default") == 0) {
125                         priv_union(deflt, pset);
126 #endif /* ! codereview */
127                 } else if (strcasecmp(offset, "none") == 0) {
128                         priv_emptyset(pset);
129                 } else if (strcasecmp(offset, "all") == 0) {
130                         priv_fillset(pset);
131                 } else if (strcasecmp(offset, "zone") == 0) {
132                         priv_union(zone, pset);
133                 } else {
134                         boolean_t neg = (*offset == '-' || *offset == '!');
135                         int privid;
136                         int slen;

138                         privid = priv_getbyname(offset +
139                             ((neg || *offset == '+') ? 1 : 0));
140                         if (privid < 0) {
141                                 slen = offset - base;
142                                 libc_free(base);
143                                 priv_freeset(pset);
144                                 if (endptr != NULL)
145                                         *endptr = priv_names + slen;
146                                 errno = EINVAL;
147                                 return (NULL);
148                         } else {
149                                 if (neg)
150                                         PRIV_DELSET(pset, privid);
151                                 else
152                                         PRIV_ADDSET(pset, privid);
153                         }
154                 }
155         }

157         libc_free(base);
158         return (pset);
159 }

161 /*
162  *      Name:   priv_set_to_str()
163  *
164  *      Description:    Given a set of privileges, list of privileges are
165  *      returned in privilege numeric order (which can be an ASCII sorted
166  *      list as our implementation allows renumbering.
167  *
168  *      String "none" identifies an empty privilege set, and string "all"
169  *      identifies a full set.
170  *
171  *      A pointer to a buffer is returned which needs to be freed by
172  *      the caller.
173  *
174  *      Several types of output are supported:
175  *              PRIV_STR_PORT           - portable output: basic,!basic
176  *              PRIV_STR_LIT            - literal output
177  *              PRIV_STR_SHORT          - shortest output
178  *
179  * NOTE: this function is called both from inside the library for the
180  * current environment and from outside the library using an externally
181  * generated priv_data_t * in order to analyze core files.  It should
182  * return strings which can be free()ed by applications and it should
183  * not use any data from the current environment except in the special
```

```
 * case that it is called from within libc, with a NULL priv_data_t *
 * argument.
 */

char *
__priv_set_to_str(
        priv_data_t *d,
        const priv_set_t *pset,
        char separator,
        int flag)
{
        const char *pstr;
        char *res, *resp;
        int i;
        char neg = separator == '!' ? '-' : '!';
        priv_set_t *zone;
        boolean_t all;
        boolean_t use_libc_data = (d == NULL);

        if (use_libc_data)
                LOADPRIVDATA(d);

        if (flag != PRIV_STR_PORT && __priv_isemptyset(d, pset))
                return (strdup("none"));
        if (flag != PRIV_STR_LIT && __priv_isfullset(d, pset))
                return (strdup("all"));

        /* Safe upper bound: global info contains all NULL separated privs */
        res = resp = alloca(d->pd_pinfo->priv_globalinfosize);

        /*
         * Compute the shortest form; i.e., the form with the fewest privilege
         * tokens.
         * The following forms are possible:
         *      literal: priv1,priv2,priv3
         *              tokcount = present
         *      port: basic,!missing_basic,other
         *              tokcount = 1 + present - presentbasic + missingbasic
         *      zone: zone,!missing_zone
         *              tokcount = 1 + missingzone
         *      all: all,!missing1,!missing2
         *              tokcount = 1 + d->pd_nprivs - present;
         *
         * Note that zone and all forms are identical in the global zone;
         * in that case (or any other where the token count is the same),
         * all is preferred.  Also, the zone form is only used when the
         * indicated privileges are a subset of the zone set.
         */

        if (use_libc_data)
                LOCKPRIVDATA();

        if (flag == PRIV_STR_SHORT) {
                int presentbasic, missingbasic, present, missing;
                int presentzone, missingzone;
                int count;

                presentbasic = missingbasic = present = 0;
                presentzone = missingzone = 0;
                zone = d->pd_zoneset;

                for (i = 0; i < d->pd_nprivs; i++) {
                        int mem = PRIV_ISMEMBER(pset, i);
                        if (d->pd_basicset != NULL &&
                            PRIV_ISMEMBER(d->pd_basicset, i)) {
                                if (mem)
```



```
                                        presentbasic++;
                                else
                                        missingbasic++;
                        }
                        if (zone != NULL && PRIV_ISMEMBER(zone, i)) {
                                if (mem)
                                        presentzone++;
                                else
                                        missingzone++;
                        }
                        if (mem)
                                present++;
                }
                missing = d->pd_nprivs - present;

                if (1 - presentbasic + missingbasic < 0) {
                        flag = PRIV_STR_PORT;
                        count = present + 1 - presentbasic + missingbasic;
                } else {
                        flag = PRIV_STR_LIT;
                        count = present;
                }
                if (count >= 1 + missing) {
                        flag = PRIV_STR_SHORT;
                        count = 1 + missing;
                        all = B_TRUE;
                }
                if (present == presentzone && 1 + missingzone < count) {
                        flag = PRIV_STR_SHORT;
                        all = B_FALSE;
                }
        }

        switch (flag) {
        case PRIV_STR_LIT:
                *res = '\0';
                break;
        case PRIV_STR_PORT:
                (void) strcpy(res, "basic");
                if (d->pd_basicset == NULL)
                        flag = PRIV_STR_LIT;
                break;
        case PRIV_STR_SHORT:
                if (all)
                        (void) strcpy(res, "all");
                else
                        (void) strcpy(res, "zone");
                break;
        default:
                if (use_libc_data)
                        UNLOCKPRIVDATA();
                return (NULL);
        }
        res += strlen(res);

        for (i = 0; i < d->pd_nprivs; i++) {
                /* Map the privilege to the next one sorted by name */
                int priv = d->pd_setsort[i];

                if (PRIV_ISMEMBER(pset, priv)) {
                        switch (flag) {
                        case PRIV_STR_SHORT:
                                if (all || PRIV_ISMEMBER(zone, priv))
                                        continue;
                                break;
                        case PRIV_STR_PORT:
```

```
316                                 if (PRIV_ISMEMBER(d->pd_basicset, priv))
317                                         continue;
318                                 break;
319                         case PRIV_STR_LIT:
320                                 break;
321                         }
322                         if (res != resp)
323                                 *res++ = separator;
324                 } else {
325                         switch (flag) {
326                         case PRIV_STR_LIT:
327                                 continue;
328                         case PRIV_STR_PORT:
329                                 if (!PRIV_ISMEMBER(d->pd_basicset, priv))
330                                         continue;
331                                 break;
332                         case PRIV_STR_SHORT:
333                                 if (!all && !PRIV_ISMEMBER(zone, priv))
334                                         continue;
335                                 break;
336                         }
337                         if (res != resp)
338                                 *res++ = separator;
339                         *res++ = neg;
340                 }
341                 pstr = __priv_getbynum(d, priv);
342                 (void) strcpy(res, pstr);
343                 res += strlen(pstr);
344         }
345         if (use_libc_data)
346                 UNLOCKPRIVDATA();
347         /* Special case the set with some high bits set */
348         return (strdup(*resp == '\0' ? "none" : resp));
349 }

351 /*
352  * priv_set_to_str() is defined to return a string that
353  * the caller must deallocate with free(3C).  Grr...
354  */
355 char *
356 priv_set_to_str(const priv_set_t *pset, char separator, int flag)
357 {
358         return (__priv_set_to_str(NULL, pset, separator, flag));
359 }

361 static char *
362 do_priv_gettext(const char *priv, const char *file)
363 {
364         char buf[8*1024];
365         boolean_t inentry = B_FALSE;
366         FILE    *namefp;

368         namefp = fopen(file, "rF");
369         if (namefp == NULL)
370                 return (NULL);

372         /*
373          * parse the file; it must have the following format
374          * Lines starting with comments "#"
375          * Lines starting with non white space with one single token:
376          * the privileges; white space indented lines which are the
377          * description; no empty lines are allowed in the description.
378          */
379         while (fgets(buf, sizeof (buf), namefp) != NULL) {
380                 char *lp;               /* pointer to the current line */
```



```
382                 if (buf[0] == '#')
383                         continue;

385                 if (buf[0] == '\n') {
386                         inentry = B_FALSE;
387                         continue;
388                 }

390                 if (inentry)
391                         continue;

393                 /* error; not skipping; yet line starts with white space */
394                 if (isspace((unsigned char)buf[0]))
395                         goto out;

397                 /* Trim trailing newline */
398                 buf[strlen(buf) - 1] = '\0';

400                 if (strcasecmp(buf, priv) != 0) {
401                         inentry = B_TRUE;
402                         continue;
403                 }

405                 lp = buf;
406                 while (fgets(lp, sizeof (buf) - (lp - buf), namefp) != NULL) {
407                         char *tstart;   /* start of text */
408                         int len;

410                         /* Empty line or start of next entry terminates */
411                         if (*lp == '\n' || !isspace((unsigned char)*lp)) {
412                                 *lp = '\0';
413                                 (void) fclose(namefp);
414                                 return (strdup(buf));
415                         }

417                         /* Remove leading white space */
418                         tstart = lp;
419                         while (*tstart != '\0' &&
420                             isspace((unsigned char)*tstart)) {
421                                 tstart++;
422                         }

424                         len = strlen(tstart);
425                         (void) memmove(lp, tstart, len + 1);
426                         lp += len;

428                         /* Entry to big; prevent fgets() loop */
429                         if (lp == &buf[sizeof (buf) - 1])
430                                 goto out;
431                 }
432                 if (lp != buf) {
433                         *lp = '\0';
434                         (void) fclose(namefp);
435                         return (strdup(buf));
436                 }
437         }
438 out:
439         (void) fclose(namefp);
440         return (NULL);
441 }

443 /*
444  * priv_gettext() is defined to return a string that
445  * the caller must deallocate with free(3C).  Grr...
446  */
447 char *
```

```
448 priv_gettext(const char *priv)
449 {
450         char file[MAXPATHLEN];
451         locale_t curloc;
452         const char *loc;
453         char    *ret;

455         /* Not a valid privilege */
456         if (priv_getbyname(priv) < 0)
457                 return (NULL);

459         curloc = uselocale(NULL);
460         loc = current_locale(curloc, LC_MESSAGES);

462         if (snprintf(file, sizeof (file),
463             _DFLT_LOC_PATH "%s/LC_MESSAGES/priv_names", loc) < sizeof (file)) {
464                 ret = do_priv_gettext(priv, (const char *)file);
465                 if (ret != NULL)
466                         return (ret);
467         }

469         /* If the path is too long or can't be opened, punt to default */
470         ret = do_priv_gettext(priv, "/etc/security/priv_names");
471         return (ret);
472 }
```

```
**********************************************************
   20827 Mon Dec 28 20:02:03 2015
new/usr/src/lib/libc/port/gen/privlib.c
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
______unchanged_portion_omitted_
```

```
137 priv_data_t *
138 __priv_parse_info(priv_impl_info_t *ip)
139 {
140         priv_data_t *tmp;
141         char *x;
142         size_t size = PRIV_IMPL_INFO_SIZE(ip);
143         int i;

145         tmp = libc_malloc(sizeof (*tmp));

147         if (tmp == NULL)
148                 return (NULL);

150         (void) memset(tmp, 0, sizeof (*tmp));

152         tmp->pd_pinfo = ip;
153         tmp->pd_setsize = sizeof (priv_chunk_t) * ip->priv_setsize;
154         tmp->pd_ucredsize = UCRED_SIZE(ip);

156         x = (char *)ip;
157         x += ip->priv_headersize;

159         while (x < ((char *)ip) + size) {
160                 /* LINTED: alignment */
161                 priv_info_names_t *na = (priv_info_names_t *)x;
162                 /* LINTED: alignment */
163                 priv_info_set_t *st = (priv_info_set_t *)x;
164                 struct strint *tmparr;

166                 switch (na->info.priv_info_type) {
167                 case PRIV_INFO_SETNAMES:
168                         if (parseninfo(na, &tmp->pd_setnames, &tmp->pd_nsets))
169                                 goto out;
170                         break;
171                 case PRIV_INFO_PRIVNAMES:
172                         if (parseninfo(na, &tmp->pd_privnames, &tmp->pd_nprivs))
173                                 goto out;
174                         /*
175                          * We compute a sorted index which allows us
176                          * to present a sorted list of privileges
177                          * without actually having to sort it each time.
178                          */
179                         tmp->pd_setsort = libc_malloc(tmp->pd_nprivs *
180                             sizeof (int));
181                         if (tmp->pd_setsort == NULL)
182                                 goto out;

184                         tmparr = libc_malloc(tmp->pd_nprivs *
185                             sizeof (struct strint));

187                         if (tmparr == NULL)
188                                 goto out;

190                         for (i = 0; i < tmp->pd_nprivs; i++) {
191                                 tmparr[i].rank = i;
192                                 tmparr[i].name = tmp->pd_privnames[i];
193                         }
194                         qsort(tmparr, tmp->pd_nprivs, sizeof (struct strint),
195                             strintcmp);
```



```
196                         for (i = 0; i < tmp->pd_nprivs; i++)
197                                 tmp->pd_setsort[i] = tmparr[i].rank;
198                         libc_free(tmparr);
199                         break;
200                 case PRIV_INFO_BASICPRIVS:
201                         tmp->pd_basicset = (priv_set_t *)&st->set[0];
202                         break;
203                 case PRIV_INFO_DEFAULTPRIVS:
204                         tmp->pd_defaultset = (priv_set_t *)&st->set[0];
205                         break;
206 #endif /* ! codereview */
207                 default:
208                         /* unknown, ignore */
209                         break;
210                 }
211                 x += na->info.priv_info_size;
212         }
213         return (tmp);
214 out:
215         libc_free(tmp->pd_setnames);
216         libc_free(tmp->pd_privnames);
217         libc_free(tmp->pd_setsort);
218         libc_free(tmp);
219         return (NULL);
220 }

222 /*
223  * Caller must have allocated d->pd_pinfo and should free it,
224  * if necessary.
225  */
226 void
227 __priv_free_info(priv_data_t *d)
228 {
229         libc_free(d->pd_setnames);
230         libc_free(d->pd_privnames);
231         libc_free(d->pd_setsort);
232         libc_free(d);
233 }

235 /*
236  * Return with the pd_lock held and data loaded or indicate failure.
237  */
238 int
239 lock_data(void)
240 {
241         if (__priv_getdata() == NULL)
242                 return (-1);

244         lmutex_lock(&pd_lock);
245         return (0);
246 }

248 boolean_t
249 refresh_data(void)
250 {
251         priv_impl_info_t *ip, ii;
252         priv_data_t *tmp;
253         char *p0, *q0;
254         int oldn, newn;
255         int i;

257         if (getprivinfo(&ii, sizeof (ii)) != 0 ||
258             ii.priv_max == privdata->pd_nprivs)
259                 return (B_FALSE);

261         ip = alloca(PRIV_IMPL_INFO_SIZE(&ii));
```

```
263         (void) getprivinfo(ip, PRIV_IMPL_INFO_SIZE(&ii));

265         /* Parse the info; then copy the additional bits */
266         tmp = __priv_parse_info(ip);
267         if (tmp == NULL)
268                 return (B_FALSE);

270         oldn = privdata->pd_nprivs;
271         p0 = privdata->pd_privnames[0];

273         newn = tmp->pd_nprivs;
274         q0 = tmp->pd_privnames[0];

276         /* copy the extra information to the old datastructure */
277         (void) memcpy((char *)privdata->pd_pinfo + sizeof (priv_impl_info_t),
278             (char *)ip + sizeof (priv_impl_info_t),
279             PRIV_IMPL_INFO_SIZE(ip) - sizeof (priv_impl_info_t));

281         /* Copy the first oldn pointers */
282         (void) memcpy(tmp->pd_privnames, privdata->pd_privnames,
283             oldn * sizeof (char *));

285         /* Adjust the rest */
286         for (i = oldn; i < newn; i++)
287                 tmp->pd_privnames[i] += p0 - q0;

289         /* Install the larger arrays */
290         libc_free(privdata->pd_privnames);
291         privdata->pd_privnames = tmp->pd_privnames;
292         tmp->pd_privnames = NULL;

294         libc_free(privdata->pd_setsort);
295         privdata->pd_setsort = tmp->pd_setsort;
296         tmp->pd_setsort = NULL;

298         /* Copy the rest of the data */
299         *privdata->pd_pinfo = *ip;

301         privdata->pd_nprivs = newn;

303         __priv_free_info(tmp);
304         return (B_TRUE);
305 }

307 void
308 unlock_data(void)
309 {
310         lmutex_unlock(&pd_lock);
311 }

313 static priv_set_t *__priv_allocset(priv_data_t *);

315 priv_data_t *
316 __priv_getdata(void)
317 {
318         if (privdata == NULL) {
319                 lmutex_lock(&pd_lock);
320                 if (privdata == NULL) {
321                         priv_data_t *tmp;
322                         priv_impl_info_t *ip;
323                         size_t size = sizeof (priv_impl_info_t) + 2048;
324                         size_t realsize;
325                         priv_impl_info_t *aip = alloca(size);

327                         if (getprivinfo(aip, size) != 0)
```



```
328                                 goto out;

330                         realsize = PRIV_IMPL_INFO_SIZE(aip);

332                         ip = libc_malloc(realsize);

334                         if (ip == NULL)
335                                 goto out;

337                         if (realsize <= size) {
338                                 (void) memcpy(ip, aip, realsize);
339                         } else if (getprivinfo(ip, realsize) != 0) {
340                                 libc_free(ip);
341                                 goto out;
342                         }

344                         if ((tmp = __priv_parse_info(ip)) == NULL) {
345                                 libc_free(ip);
346                                 goto out;
347                         }

349                         /* Allocate the zoneset just once, here */
350                         tmp->pd_zoneset = __priv_allocset(tmp);
351                         if (tmp->pd_zoneset == NULL)
352                                 goto clean;

354                         if (zone_getattr(getzoneid(), ZONE_ATTR_PRIVSET,
355                             tmp->pd_zoneset, tmp->pd_setsize)
356                             == tmp->pd_setsize) {
357                                 membar_producer();
358                                 privdata = tmp;
359                                 goto out;
360                         }

362                         priv_freeset(tmp->pd_zoneset);
363 clean:
364                         __priv_free_info(tmp);
365                         libc_free(ip);
366                 }
367 out:
368                 lmutex_unlock(&pd_lock);
369         }
370         membar_consumer();
371         return (privdata);
372 }

374 const priv_impl_info_t *
375 getprivimplinfo(void)
376 {
377         priv_data_t *d;

379         LOADPRIVDATA(d);

381         return (d->pd_pinfo);
382 }

384 static priv_set_t *
385 priv_vlist(va_list ap)
386 {
387         priv_set_t *pset = priv_allocset();
388         const char *priv;

390         if (pset == NULL)
391                 return (NULL);

393         priv_emptyset(pset);
```

```
395         while ((priv = va_arg(ap, const char *)) != NULL) {
396                 if (priv_addset(pset, priv) < 0) {
397                         priv_freeset(pset);
398                         return (NULL);
399                 }
400         }
401         return (pset);
402 }

404 /*
405  * priv_set(op, set, priv_id1, priv_id2, ..., NULL)
406  *
407  * Library routine to enable a user process to set a specific
408  * privilege set appropriately using a single call.  User is
409  * required to terminate the list of privileges with NULL.
410  */
411 int
412 priv_set(priv_op_t op, priv_ptype_t setname, ...)
413 {
414         va_list ap;
415         priv_set_t *pset;
416         int ret;

418         va_start(ap, setname);

420         pset = priv_vlist(ap);

422         va_end(ap);

424         if (pset == NULL)
425                 return (-1);

427         /* All sets */
428         if (setname == NULL) {
429                 priv_data_t *d;
430                 int set;

432                 LOADPRIVDATA(d);

434                 for (set = 0; set < d->pd_nsets; set++)
435                         if ((ret = syscall(SYS_privsys, PRIVSYS_SETPPRIV, op,
436                             set, (void *)pset, d->pd_setsize)) != 0)
437                                 break;
438         } else {
439                 ret = setppriv(op, setname, pset);
440         }

442         priv_freeset(pset);
443         return (ret);
444 }

446 /*
447  * priv_ineffect(privilege).
448  * tests the existence of a privilege against the effective set.
449  */
450 boolean_t
451 priv_ineffect(const char *priv)
452 {
453         priv_set_t *curset;
454         boolean_t res;

456         curset = priv_allocset();

458         if (curset == NULL)
459                 return (B_FALSE);
```



```
461         if (getppriv(effective, curset) != 0 ||
462             !priv_ismember(curset, priv))
463                 res = B_FALSE;
464         else
465                 res = B_TRUE;

467         priv_freeset(curset);

469         return (res);
470 }

472 /*
473  * The routine __init_daemon_priv() is private to Solaris and is
474  * used by daemons to limit the privileges they can use and
475  * to set the uid they run under.
476  */

478 static const char root_cp[] = "/core.%f.%t";
479 static const char daemon_cp[] = "/var/tmp/core.%f.%t";

481 int
482 __init_daemon_priv(int flags, uid_t uid, gid_t gid, ...)
483 {
484         priv_set_t *nset;
485         priv_set_t *perm = NULL;
486         va_list pa;
487         priv_data_t *d;
488         int ret = -1;
489         char buf[1024];

491         LOADPRIVDATA(d);

493         va_start(pa, gid);

495         nset = priv_vlist(pa);

497         va_end(pa);

499         if (nset == NULL)
500                 return (-1);

502         /* Always add the basic set */
503         /* XXX: Always add the _default_ set? */
504 #endif /* ! codereview */
505         if (d->pd_basicset != NULL)
506                 priv_union(d->pd_basicset, nset);

508         /*
509          * This is not a significant failure: it allows us to start programs
510          * with sufficient privileges and with the proper uid.   We don't
511          * care enough about the extra groups in that case.
512          */
513         if (flags & PU_RESETGROUPS)
514                 (void) setgroups(0, NULL);

516         if (gid != (gid_t)-1 && setgid(gid) != 0)
517                 goto end;

519         perm = priv_allocset();
520         if (perm == NULL)
521                 goto end;

523         /* E = P */
524         (void) getppriv(permitted, perm);
525         (void) setppriv(PRIV_SET, effective, perm);
```

```
527         /* Now reset suid and euid */
528         if (uid != (uid_t)-1 && setreuid(uid, uid) != 0)
529                 goto end;

531         /* Check for the limit privs */
532         if ((flags & PU_LIMITPRIVS) &&
533             setppriv(PRIV_SET, limit, nset) != 0)
534                 goto end;

536         if (flags & PU_CLEARLIMITSET) {
537                 priv_emptyset(perm);
538                 if (setppriv(PRIV_SET, limit, perm) != 0)
539                         goto end;
540         }

542         /* Remove the privileges from all the other sets */
543         if (setppriv(PRIV_SET, permitted, nset) != 0)
544                 goto end;

546         if (!(flags & PU_INHERITPRIVS))
547                 priv_emptyset(nset);

549         ret = setppriv(PRIV_SET, inheritable, nset);
550 end:
551         priv_freeset(nset);
552         priv_freeset(perm);

554         if (core_get_process_path(buf, sizeof (buf), getpid()) == 0 &&
555             strcmp(buf, "core") == 0) {

557                 if ((uid == (uid_t)-1 ? geteuid() : uid) == 0) {
558                         (void) core_set_process_path(root_cp, sizeof (root_cp),
559                             getpid());
560                 } else {
561                         (void) core_set_process_path(daemon_cp,
562                             sizeof (daemon_cp), getpid());
563                 }
564         }
565         (void) setpflags(__PROC_PROTECT, 0);

567         return (ret);
568 }

570 /*
571  * The routine __fini_daemon_priv() is private to Solaris and is
572  * used by daemons to clear remaining unwanted privileges and
573  * reenable core dumps.
574  */
575 void
576 __fini_daemon_priv(const char *priv, ...)
577 {
578         priv_set_t *nset;
579         va_list pa;

581         if (priv != NULL) {

583                 va_start(pa, priv);
584                 nset = priv_vlist(pa);
585                 va_end(pa);

587                 if (nset == NULL)
588                         return;

590                 (void) priv_addset(nset, priv);
591                 (void) setppriv(PRIV_OFF, permitted, nset);
```



```
592                 priv_freeset(nset);
593         }

595         (void) setpflags(__PROC_PROTECT, 0);
596 }

598 /*
599  * The routine __init_suid_priv() is private to Solaris and is
600  * used by set-uid root programs to limit the privileges acquired
601  * to those actually needed.
602  */

604 static priv_set_t *bracketpriv;

606 int
607 __init_suid_priv(int flags, ...)
608 {
609         priv_set_t *nset = NULL;
610         priv_set_t *tmpset = NULL;
611         va_list pa;
612         int r = -1;
613         uid_t ruid, euid;

615         euid = geteuid();

617         /* If we're not set-uid root, don't reset the uid */
618         if (euid == 0) {
619                 ruid = getuid();
620                 /* If we're running as root, keep everything */
621                 if (ruid == 0)
622                         return (0);
623         }

625         /* Can call this only once */
626         if (bracketpriv != NULL)
627                 return (-1);

629         va_start(pa, flags);

631         nset = priv_vlist(pa);

633         va_end(pa);

635         if (nset == NULL)
636                 goto end;

638         tmpset = priv_allocset();

640         if (tmpset == NULL)
641                 goto end;

643         /* We cannot grow our privileges beyond P, so start there */
644         (void) getppriv(permitted, tmpset);

646         /* Is the privilege we need even in P? */
647         if (!priv_issubset(nset, tmpset))
648                 goto end;

650         bracketpriv = priv_allocset();
651         if (bracketpriv == NULL)
652                 goto end;

654         priv_copyset(nset, bracketpriv);

656         /* Always add the basic set */
657         /* XXX: Always add the default set? */
```

```
#endif /* ! codereview */
        priv_union(priv_basic(), nset);

        /* But don't add what we don't have */
        priv_intersect(tmpset, nset);

        (void) getppriv(inheritable, tmpset);

        /* And stir in the inheritable privileges */
        priv_union(tmpset, nset);

        if ((r = setppriv(PRIV_SET, effective, tmpset)) != 0)
                goto end;

        if ((r = setppriv(PRIV_SET, permitted, nset)) != 0)
                goto end;

        if (flags & PU_CLEARLIMITSET)
                priv_emptyset(nset);

        if ((flags & (PU_LIMITPRIVS|PU_CLEARLIMITSET)) != 0 &&
            (r = setppriv(PRIV_SET, limit, nset)) != 0)
                goto end;

        if (euid == 0)
                r = setreuid(ruid, ruid);

end:
        priv_freeset(tmpset);
        priv_freeset(nset);
        if (r != 0) {
                /* Fail without leaving uid 0 around */
                if (euid == 0)
                        (void) setreuid(ruid, ruid);
                priv_freeset(bracketpriv);
                bracketpriv = NULL;
        }

        return (r);
}

/*
 * Toggle privileges on/off in the effective set.
 */
int
__priv_bracket(priv_op_t op)
{
        /* We're running fully privileged or didn't check errors first time */
        if (bracketpriv == NULL)
                return (0);

        /* Only PRIV_ON and PRIV_OFF are valid */
        if (op == PRIV_SET)
                return (-1);

        return (setppriv(op, effective, bracketpriv));
}

/*
 * Remove privileges from E & P.
 */
void
__priv_relinquish(void)
{
        if (bracketpriv != NULL) {
                (void) setppriv(PRIV_OFF, permitted, bracketpriv);
```



```
                priv_freeset(bracketpriv);
                bracketpriv = NULL;
        }
}

/*
 * Use binary search on the ordered list.
 */
int
__priv_getbyname(const priv_data_t *d, const char *name)
{
        char *const *list;
        const int *order;
        int lo = 0;
        int hi;

        if (d == NULL)
                return (-1);

        list = d->pd_privnames;
        order = d->pd_setsort;
        hi = d->pd_nprivs - 1;

        if (strncasecmp(name, "priv_", 5) == 0)
                name += 5;

        do {
                int mid = (lo + hi) / 2;
                int res = strcasecmp(name, list[order[mid]]);

                if (res == 0)
                        return (order[mid]);
                else if (res < 0)
                        hi = mid - 1;
                else
                        lo = mid + 1;
        } while (lo <= hi);

        errno = EINVAL;
        return (-1);
}

int
priv_getbyname(const char *name)
{
        WITHPRIVLOCKED(int, -1, __priv_getbyname(GETPRIVDATA(), name))
}

int
__priv_getsetbyname(const priv_data_t *d, const char *name)
{
        int i;
        int n = d->pd_nsets;
        char *const *list = d->pd_setnames;

        if (strncasecmp(name, "priv_", 5) == 0)
                name += 5;

        for (i = 0; i < n; i++) {
                if (strcasecmp(list[i], name) == 0)
                        return (i);
        }

        errno = EINVAL;
        return (-1);
}
```

```
791 int
792 priv_getsetbyname(const char *name)
793 {
794         /* Not locked: sets don't change */
795         return (__priv_getsetbyname(GETPRIVDATA(), name));
796 }

798 static const char *
799 priv_bynum(int i, int n, char **list)
800 {
801         if (i < 0 || i >= n)
802                 return (NULL);

804         return (list[i]);
805 }

807 const char *
808 __priv_getbynum(const priv_data_t *d, int num)
809 {
810         if (d == NULL)
811                 return (NULL);
812         return (priv_bynum(num, d->pd_nprivs, d->pd_privnames));
813 }

815 const char *
816 priv_getbynum(int num)
817 {
818         WITHPRIVLOCKED(const char *, NULL, __priv_getbynum(GETPRIVDATA(), num))
819 }

821 const char *
822 __priv_getsetbynum(const priv_data_t *d, int num)
823 {
824         if (d == NULL)
825                 return (NULL);
826         return (priv_bynum(num, d->pd_nsets, d->pd_setnames));
827 }

829 const char *
830 priv_getsetbynum(int num)
831 {
832         return (__priv_getsetbynum(GETPRIVDATA(), num));
833 }


836 /*
837  * Privilege manipulation functions
838  *
839  * Without knowing the details of the privilege set implementation,
840  * opaque pointers can be used to manipulate sets at will.
841  */

843 static priv_set_t *
844 __priv_allocset(priv_data_t *d)
845 {
846         if (d == NULL)
847                 return (NULL);

849         return (libc_malloc(d->pd_setsize));
850 }

852 priv_set_t *
853 priv_allocset(void)
854 {
855         return (__priv_allocset(GETPRIVDATA()));
```



```
856 }

858 void
859 priv_freeset(priv_set_t *p)
860 {
861         int er = errno;

863         libc_free(p);
864         errno = er;
865 }

867 void
868 __priv_emptyset(priv_data_t *d, priv_set_t *set)
869 {
870         (void) memset(set, 0, d->pd_setsize);
871 }

873 void
874 priv_emptyset(priv_set_t *set)
875 {
876         __priv_emptyset(GETPRIVDATA(), set);
877 }

879 void
880 priv_basicset(priv_set_t *set)
881 {
882         priv_copyset(priv_basic(), set);
883 }

885 void
886 priv_defaultset(priv_set_t *set)
887 {
888         priv_copyset(priv_default(), set);
889 }

891 void
892 #endif /* ! codereview */
893 __priv_fillset(priv_data_t *d, priv_set_t *set)
894 {
895         (void) memset(set, ~0, d->pd_setsize);
896 }

898 void
899 priv_fillset(priv_set_t *set)
900 {
901         __priv_fillset(GETPRIVDATA(), set);
902 }


905 #define PRIV_TEST_BODY_D(d, test) \
906         int i; \
907 \
908         for (i = d->pd_pinfo->priv_setsize; i-- > 0; ) \
909                 if (!(test)) \
910                         return (B_FALSE); \
911 \
912         return (B_TRUE)

914 boolean_t
915 priv_isequalset(const priv_set_t *a, const priv_set_t *b)
916 {
917         priv_data_t *d;

919         LOADPRIVDATA(d);

921         return ((boolean_t)(memcmp(a, b, d->pd_setsize) == 0));
```

```
922 }

924 boolean_t
925 __priv_isemptyset(priv_data_t *d, const priv_set_t *set)
926 {
927         PRIV_TEST_BODY_D(d, ((priv_chunk_t *)set)[i] == 0);
928 }

930 boolean_t
931 priv_isemptyset(const priv_set_t *set)
932 {
933         return (__priv_isemptyset(GETPRIVDATA(), set));
934 }

936 boolean_t
937 __priv_isfullset(priv_data_t *d, const priv_set_t *set)
938 {
939         PRIV_TEST_BODY_D(d, ((priv_chunk_t *)set)[i] == ~(priv_chunk_t)0);
940 }

942 boolean_t
943 priv_isfullset(const priv_set_t *set)
944 {
945         return (__priv_isfullset(GETPRIVDATA(), set));
946 }

948 /*
949  * Return true if a is a subset of b
950  */
951 boolean_t
952 __priv_issubset(priv_data_t *d, const priv_set_t *a, const priv_set_t *b)
953 {
954         PRIV_TEST_BODY_D(d, (((priv_chunk_t *)a)[i] | ((priv_chunk_t *)b)[i]) ==
955             ((priv_chunk_t *)b)[i]);
956 }

958 boolean_t
959 priv_issubset(const priv_set_t *a, const priv_set_t *b)
960 {
961         return (__priv_issubset(GETPRIVDATA(), a, b));
962 }

964 #define PRIV_CHANGE_BODY(a, op, b) \
965         int i; \
966         priv_data_t *d; \
967 \
968         LOADPRIVDATA(d); \
969 \
970         for (i = 0; i < d->pd_pinfo->priv_setsize; i++) \
971                 ((priv_chunk_t *)a)[i] op \
972                         ((priv_chunk_t *)b)[i]

974 /* B = A ^ B */
975 void
976 priv_intersect(const priv_set_t *a, priv_set_t *b)
977 {
978         /* CSTYLED */
979         PRIV_CHANGE_BODY(b, &=, a);
980 }

982 /* B = A */
983 void
984 priv_copyset(const priv_set_t *a, priv_set_t *b)
985 {
986         /* CSTYLED */
987         PRIV_CHANGE_BODY(b, =, a);
```



```
988 }

990 /* B = A v B */
991 void
992 priv_union(const priv_set_t *a, priv_set_t *b)
993 {
994         /* CSTYLED */
995         PRIV_CHANGE_BODY(b, |=, a);
996 }

998 /* A = ! A */
999 void
1000 priv_inverse(priv_set_t *a)
1001 {
1002         PRIV_CHANGE_BODY(a, = ~, a);
1003 }

1005 /*
1006  * Manipulating single privileges.
1007  */

1009 int
1010 priv_addset(priv_set_t *a, const char *p)
1011 {
1012         int priv = priv_getbyname(p);

1014         if (priv < 0)
1015                 return (-1);

1017         PRIV_ADDSET(a, priv);

1019         return (0);
1020 }

1022 int
1023 priv_delset(priv_set_t *a, const char *p)
1024 {
1025         int priv = priv_getbyname(p);

1027         if (priv < 0)
1028                 return (-1);

1030         PRIV_DELSET(a, priv);
1031         return (0);
1032 }

1034 boolean_t
1035 priv_ismember(const priv_set_t *a, const char *p)
1036 {
1037         int priv = priv_getbyname(p);

1039         if (priv < 0)
1040                 return (B_FALSE);

1042         return ((boolean_t)PRIV_ISMEMBER(a, priv));
1043 }
```

```
**********************************************************
   57674 Mon Dec 28 20:02:08 2015
new/usr/src/lib/libc/port/mapfile-vers
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
```

```
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
# Copyright (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#
# Copyright 2010 Nexenta Systems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# Copyright (c) 2012 by Delphix. All rights reserved.
# Copyright (c) 2015, Joyent, Inc.  All rights reserved.
# Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
# Copyright (c) 2013 Gary Mills
# Copyright 2014 Garrett D'Amore <garrett@damore.org>

#
# MAPFILE HEADER START
#
# WARNING:  STOP NOW.  DO NOT MODIFY THIS FILE.
# Object versioning must comply with the rules detailed in
#
#       usr/src/lib/README.mapfiles
#
# You should not be making modifications here until you've read the most current
# copy of that file. If you need help, contact a gatekeeper for guidance.
#
# MAPFILE HEADER END
#

$mapfile_version 2

#
# All function names added to this or any other libc mapfile
# must be placed under the 'protected:' designation.
# The 'global:' designation is used *only* for data
# items and for the members of the malloc() family.
#

#
# README README README README README README: how to update this file
#   1) each version of Solaris/OpenSolaris gets a version number.
#      (Actually since Solaris is actually a series of OpenSolaris releases
#       we'll just use OpenSolaris for this exercise.)
#       OpenSolaris 2008.11 gets 1.23
```



```
#       OpenSolaris 2009.04 gets 1.24
#       etc.
#   2) each project integration uses a unique version number.
#       PSARC/2008/123 gets 1.24.1
#       PSARC/2008/456 gets 1.24.2
#       etc.
#


# Mnemonic conditional input identifiers:
#
# - amd64, i386, sparc32, sparcv9: Correspond to ISA subdirectories used to
#       hold per-platform code. Note however that we use 'sparc32' instead of
#       'sparc'. Since '_sparc' is predefined to apply to, all sparc platforms,
#       naming the 32-bit version 'sparc' would be too likely to cause errors.
#
# -     lf64: Defined on platforms that offer the 32-bit largefile APIs
#
$if _ELF32
$add lf64
$endif
$if _sparc && _ELF32
$add sparc32
$endif
$if _sparc && _ELF64
$add sparcv9
$endif
$if _x86 && _ELF32
$add i386
$endif
$if _x86 && _ELF64
$add amd64
$endif

SYMBOL_VERSION ILLUMOS_0.20 {   # priv_defaultset
        priv_defaultset;
} ILLUMOS_0.19;

#endif /* ! codereview */
SYMBOL_VERSION ILLUMOS_0.19 {   # flock
    protected:
        flock;
} ILLUMOS_0.18;

SYMBOL_VERSION ILLUMOS_0.18 {   # signalfd
    protected:
        signalfd;
} ILLUMOS_0.17;

SYMBOL_VERSION ILLUMOS_0.17 {   # glob(3C) LFS
$if lf64
    protected:
        _glob_ext64;
        _globfree_ext64;
$endif
} ILLUMOS_0.16;

SYMBOL_VERSION ILLUMOS_0.16 {   # timerfd
    protected:
        timerfd_create;
        timerfd_gettime;
        timerfd_settime;
} ILLUMOS_0.15;

SYMBOL_VERSION ILLUMOS_0.15 {   # epoll(3C)
    protected:
```

```
128         epoll_create;
129         epoll_create1;
130         epoll_ctl;
131         epoll_wait;
132         epoll_pwait;
133 } ILLUMOS_0.14;

135 SYMBOL_VERSION ILLUMOS_0.14 {   # strerror_l
136     protected:
137         strerror_l;
138 } ILLUMOS_0.13;

140 SYMBOL_VERSION ILLUMOS_0.13 {   # eventfd
141     protected:
142         eventfd;
143         eventfd_read;
144         eventfd_write;
145 } ILLUMOS_0.12;

147 SYMBOL_VERSION ILLUMOS_0.12 {   # arc4random and friends
148     protected:
149         arc4random;
150         arc4random_buf;
151         arc4random_uniform;
152         explicit_bzero;
153         getentropy;
154 } ILLUMOS_0.11;

156 SYMBOL_VERSION ILLUMOS_0.11 {   # Illumos additions
157     protected:
158         iswxdigit_l;
159         isxdigit_l;
160 } ILLUMOS_0.10;

162 SYMBOL_VERSION ILLUMOS_0.10 {   # Illumos additions
163     protected:
164         preadv;
165         pwritev;

167 $if lf64
168         preadv64;
169         pwritev64;
170 $endif
171 } ILLUMOS_0.9;

173 SYMBOL_VERSION ILLUMOS_0.9 {
174     protected:
175         wcsnrtombs;
176         wcsnrtombs_l;
177 } ILLUMOS_0.8;

179 SYMBOL_VERSION ILLUMOS_0.8 {    # POSIX 2008 newlocale and friends
180     protected:
181         __global_locale;
182         __mb_cur_max;
183         __mb_cur_max_l;
184         btowc_l;
185         duplocale;
186         fgetwc_l;
187         freelocale;
188         getwc_l;
189         isalnum_l;
190         isalpha_l;
191         isblank_l;
192         iscntrl_l;
193         isdigit_l;
```



```
194         isgraph_l;
195         islower_l;
196         isprint_l;
197         ispunct_l;
198         isspace_l;
199         isupper_l;
200         iswideogram;
201         iswideogram_l;
202         iswnumber;
203         iswnumber_l;
204         iswhexnumber;
205         iswhexnumber_l;
206         iswphonogram;
207         iswphonogram_l;
208         iswspecial;
209         iswspecial_l;
210         iswalnum_l;
211         iswalpha_l;
212         iswblank_l;
213         iswcntrl_l;
214         iswctype_l;
215         iswdigit_l;
216         iswgraph_l;
217         iswlower_l;
218         iswprint_l;
219         iswpunct_l;
220         iswspace_l;
221         iswupper_l;
222         mblen_l;
223         mbrlen_l;
224         mbsinit_l;
225         mbsnrtowcs;
226         mbsnrtowcs_l;
227         mbsrtowcs_l;
228         mbstowcs_l;
229         mbtowc_l;
230         newlocale;
231         nl_langinfo_l;
232         strcasecmp_l;
233         strcasestr_l;
234         strcoll_l;
235         strfmon_l;
236         strftime_l;
237         strncasecmp_l;
238         strptime_l;
239         strxfrm_l;
240         tolower_l;
241         toupper_l;
242         towlower_l;
243         towupper_l;
244         towctrans_l;
245         uselocale;
246         wcrtomb_l;
247         wcscasecmp_l;
248         wcscoll_l;
249         wcsncasecmp_l;
250         wcsrtombs_l;
251         wcstombs_l;
252         wcswidth_l;
253         wcsxfrm_l;
254         wctob_l;
255         wctomb_l;
256         wctrans_l;
257         wctype_l;
258         wcwidth_l;
259 } ILLUMOS_0.7;
```

```
SYMBOL_VERSION ILLUMOS_0.7 {    # Illumos additions
    protected:
        _glob_ext;
        _globfree_ext;
} ILLUMOS_0.6;

SYMBOL_VERSION ILLUMOS_0.6 {    # Illumos additions
    protected:
        getloginx;
        getloginx_r;
        __posix_getloginx_r;
} ILLUMOS_0.5;

SYMBOL_VERSION ILLUMOS_0.5 {    # common C++ ABI exit handlers
    protected:
        __cxa_atexit;
        __cxa_finalize;
} ILLUMOS_0.4;

SYMBOL_VERSION ILLUMOS_0.4 {    # Illumos additions
    protected:
        pipe2;
        dup3;
        mkostemp;
        mkostemps;

$if lf64
        mkostemp64;
        mkostemps64;
$endif
} ILLUMOS_0.3;

SYMBOL_VERSION ILLUMOS_0.3 {    # Illumos additions
    protected:
        assfail3;
} ILLUMOS_0.2;

SYMBOL_VERSION ILLUMOS_0.2 {    # Illumos additions
    protected:
        posix_spawn_pipe_np;
} ILLUMOS_0.1;

SYMBOL_VERSION ILLUMOS_0.1 {    # Illumos additions
    protected:
        timegm;
} SUNW_1.23;

SYMBOL_VERSION SUNW_1.23 {      # SunOS 5.11 (Solaris 11)
    global:
        _nl_domain_bindings;
        _nl_msg_cat_cntr;

$if _ELF32
        dl_iterate_phdr { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
$elif sparcv9
        dl_iterate_phdr { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
$elif amd64
        dl_iterate_phdr { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
$else
$error unknown platform
$endif

    protected:

$if sparc32
```



```
        __align_cpy_1;
$endif

        addrtosymstr;
        aio_cancel;
        aiocancel;
        aio_error;
        aio_fsync;
        aio_read;
        aioread;
        aio_return;
        aio_suspend;
        aiowait;
        aio_waitn;
        aio_write;
        aiowrite;
        asprintf;
        assfail;
        backtrace;
        backtrace_symbols;
        backtrace_symbols_fd;
        canonicalize_file_name;
        clearenv;
        clock_getres;
        clock_gettime;
        clock_nanosleep;
        clock_settime;
        daemon;
        dirfd;
        door_bind;
        door_call;
        door_create;
        door_cred;
        door_getparam;
        door_info;
        door_return;
        door_revoke;
        door_server_create;
        door_setparam;
        door_ucred;
        door_unbind;
        door_xcreate;
        err;
        errx;
        faccessat;
        fchmodat;
        fcloseall;
        fdatasync;
        ffsl;
        ffsll;
        fgetattr;
        fls;
        flsl;
        flsll;
        forkallx;
        forkx;
        fsetattr;
        getattrat;
        getdelim;
        getline;
        get_nprocs;
        get_nprocs_conf;
        getprogname;
        htonl;
        htonll;
        htons;
```

```
392         linkat;
393         lio_listio;
394         memmem;
395         mkdirat;
396         mkdtemp;
397         mkfifoat;
398         mknodat;
399         mkstemps;
400         mmapobj;
401         mq_close;
402         mq_getattr;
403         mq_notify;
404         mq_open;
405         mq_receive;
406         mq_reltimedreceive_np;
407         mq_reltimedsend_np;
408         mq_send;
409         mq_setattr;
410         mq_timedreceive;
411         mq_timedsend;
412         mq_unlink;
413         nanosleep;
414         ntohl;
415         ntohll;
416         ntohs;
417         posix_fadvise;
418         posix_fallocate;
419         posix_madvise;
420         posix_memalign;
421         posix_spawn_file_actions_addclosefrom_np;
422         posix_spawnattr_getsigignore_np;
423         posix_spawnattr_setsigignore_np;
424         ppoll;
425         priv_basicset;
426         pthread_key_create_once_np;
427         pthread_mutexattr_getrobust;
428         pthread_mutexattr_setrobust;
429         pthread_mutex_consistent;
430         readlinkat;
431         sched_getparam;
432         sched_get_priority_max;
433         sched_get_priority_min;
434         sched_getscheduler;
435         sched_rr_get_interval;
436         sched_setparam;
437         sched_setscheduler;
438         sched_yield;
439         sem_close;
440         sem_destroy;
441         sem_getvalue;
442         sem_init;
443         sem_open;
444         sem_post;
445         sem_reltimedwait_np;
446         sem_timedwait;
447         sem_trywait;
448         sem_unlink;
449         sem_wait;
450         setattrat;
451         setprogname;
452         _sharefs;
453         shm_open;
454         shm_unlink;
455         sigqueue;
456         sigtimedwait;
457         sigwaitinfo;
```



```
458         smt_pause;
459         stpcpy;
460         stpncpy;
461         strcasestr;
462         strchrnul;
463         strndup;
464         strnlen;
465         strnstr;
466         strsep;
467         symlinkat;
468         thr_keycreate_once;
469         timer_create;
470         timer_delete;
471         timer_getoverrun;
472         timer_gettime;
473         timer_settime;
474         u8_strcmp;
475         u8_validate;
476         uconv_u16tou32;
477         uconv_u16tou8;
478         uconv_u32tou16;
479         uconv_u32tou8;
480         uconv_u8tou16;
481         uconv_u8tou32;
482         vasprintf;
483         verr;
484         verrx;
485         vforkx;
486         vwarn;
487         vwarnx;
488         warn;
489         warnx;
490         wcpcpy;
491         wcpncpy;
492         wcscasecmp;
493         wcsdup;
494         wcsncasecmp;
495         wcsnlen;

497 $if lf64
498         aio_cancel64;
499         aio_error64;
500         aio_fsync64;
501         aio_read64;
502         aioread64;
503         aio_return64;
504         aio_suspend64;
505         aio_waitn64;
506         aio_write64;
507         aiowrite64;
508         lio_listio64;
509         mkstemps64;
510         posix_fadvise64;
511         posix_fallocate64;
512 $endif
513 } SUNW_1.22.6;

515 SYMBOL_VERSION SUNW_1.22.6 {    # s10u9 - SunOS 5.10 (Solaris 10) patch addition
516     protected:
517         futimens;
518         utimensat;
519 } SUNW_1.22.5;

521 SYMBOL_VERSION SUNW_1.22.5 {    # s10u8 - SunOS 5.10 (Solaris 10) patch addition
522     protected:
523         getpagesizes2;
```

```
} SUNW_1.22.4;

SYMBOL_VERSION SUNW_1.22.4 {    # s10u7 - SunOS 5.10 (Solaris 10) patch addition
    protected:
        SUNW_1.22.4;
} SUNW_1.22.3;

SYMBOL_VERSION SUNW_1.22.3 {    # SunOS 5.10 (Solaris 10) patch additions
    protected:
        mutex_consistent;
        u8_textprep_str;
        uucopy;
        uucopystr;
} SUNW_1.22.2;

SYMBOL_VERSION SUNW_1.22.2 {    # SunOS 5.10 (Solaris 10) patch additions
    protected:
        is_system_labeled;
        ucred_getlabel;
        _ucred_getlabel;
} SUNW_1.22.1;

SYMBOL_VERSION SUNW_1.22.1 {    # SunOS 5.10 (Solaris 10) patch additions
    protected:
        atomic_add_8;
        atomic_add_8_nv;
        atomic_add_char         { FLAGS = NODYNSORT };
        atomic_add_char_nv      { FLAGS = NODYNSORT };
        atomic_add_int          { FLAGS = NODYNSORT };
        atomic_add_int_nv       { FLAGS = NODYNSORT };
        atomic_add_ptr          { FLAGS = NODYNSORT };
        atomic_add_ptr_nv       { FLAGS = NODYNSORT };
        atomic_add_short        { FLAGS = NODYNSORT };
        atomic_add_short_nv     { FLAGS = NODYNSORT };
        atomic_and_16;
        atomic_and_16_nv;
        atomic_and_32_nv;
        atomic_and_64;
        atomic_and_64_nv;
        atomic_and_8;
        atomic_and_8_nv;
        atomic_and_uchar        { FLAGS = NODYNSORT };
        atomic_and_uchar_nv     { FLAGS = NODYNSORT };
        atomic_and_uint_nv      { FLAGS = NODYNSORT };
        atomic_and_ulong        { FLAGS = NODYNSORT };
        atomic_and_ulong_nv     { FLAGS = NODYNSORT };
        atomic_and_ushort       { FLAGS = NODYNSORT };
        atomic_and_ushort_nv    { FLAGS = NODYNSORT };
        atomic_cas_16;
        atomic_cas_32;
        atomic_cas_64;
        atomic_cas_8;
        atomic_cas_ptr          { FLAGS = NODYNSORT };
        atomic_cas_uchar        { FLAGS = NODYNSORT };
        atomic_cas_uint         { FLAGS = NODYNSORT };
        atomic_cas_ulong        { FLAGS = NODYNSORT };
        atomic_cas_ushort       { FLAGS = NODYNSORT };
        atomic_clear_long_excl  { FLAGS = NODYNSORT };
        atomic_dec_16;
        atomic_dec_16_nv;
        atomic_dec_32;
        atomic_dec_32_nv;
        atomic_dec_64;
        atomic_dec_64_nv;
        atomic_dec_8;
        atomic_dec_8_nv;
```



```
        atomic_dec_uchar        { FLAGS = NODYNSORT };
        atomic_dec_uchar_nv     { FLAGS = NODYNSORT };
        atomic_dec_uint         { FLAGS = NODYNSORT };
        atomic_dec_uint_nv      { FLAGS = NODYNSORT };
        atomic_dec_ulong        { FLAGS = NODYNSORT };
        atomic_dec_ulong_nv     { FLAGS = NODYNSORT };
        atomic_dec_ushort       { FLAGS = NODYNSORT };
        atomic_dec_ushort_nv    { FLAGS = NODYNSORT };
        atomic_inc_16;
        atomic_inc_16_nv;
        atomic_inc_32;
        atomic_inc_32_nv;
        atomic_inc_64;
        atomic_inc_64_nv;
        atomic_inc_8;
        atomic_inc_8_nv;
        atomic_inc_uchar        { FLAGS = NODYNSORT };
        atomic_inc_uchar_nv     { FLAGS = NODYNSORT };
        atomic_inc_uint         { FLAGS = NODYNSORT };
        atomic_inc_uint_nv      { FLAGS = NODYNSORT };
        atomic_inc_ulong        { FLAGS = NODYNSORT };
        atomic_inc_ulong_nv     { FLAGS = NODYNSORT };
        atomic_inc_ushort       { FLAGS = NODYNSORT };
        atomic_inc_ushort_nv    { FLAGS = NODYNSORT };
        atomic_or_16;
        atomic_or_16_nv;
        atomic_or_32_nv;
        atomic_or_64;
        atomic_or_64_nv;
        atomic_or_8;
        atomic_or_8_nv;
        atomic_or_uchar         { FLAGS = NODYNSORT };
        atomic_or_uchar_nv      { FLAGS = NODYNSORT };
        atomic_or_uint_nv       { FLAGS = NODYNSORT };
        atomic_or_ulong         { FLAGS = NODYNSORT };
        atomic_or_ulong_nv      { FLAGS = NODYNSORT };
        atomic_or_ushort        { FLAGS = NODYNSORT };
        atomic_or_ushort_nv     { FLAGS = NODYNSORT };
        atomic_set_long_excl    { FLAGS = NODYNSORT };
        atomic_swap_16;
        atomic_swap_32;
        atomic_swap_64;
        atomic_swap_8;
        atomic_swap_ptr         { FLAGS = NODYNSORT };
        atomic_swap_uchar       { FLAGS = NODYNSORT };
        atomic_swap_uint        { FLAGS = NODYNSORT };
        atomic_swap_ulong       { FLAGS = NODYNSORT };
        atomic_swap_ushort      { FLAGS = NODYNSORT };
        membar_consumer;
        membar_enter;
        membar_exit;
        membar_producer;

$if _ELF32
        enable_extended_FILE_stdio;
$endif

$if i386
        # Note: atomic_[and,dec,inc,or]_64_nv are also defined above. Here,
        # we add the NODYNSORT attribute to them. On this platform, they are
        # aliases for the non-_nv versions. If that is changed, these lines
        # should be removed.
        atomic_and_64_nv        { FLAGS = NODYNSORT };
        atomic_dec_64_nv        { FLAGS = NODYNSORT };
        atomic_inc_64_nv        { FLAGS = NODYNSORT };
        atomic_or_64_nv         { FLAGS = NODYNSORT };
```

```
$endif
$if _sparc
        # Note: atomic_OP_WIDTH_nv symbols are also defined above. Here,
        # we add the NODYNSORT attribute to them. On this platform, they are
        # aliases for the non-_nv versions. If that is changed, these lines
        # should be removed.
        atomic_add_8_nv         { FLAGS = NODYNSORT };
        atomic_and_8_nv         { FLAGS = NODYNSORT };
        atomic_and_16_nv        { FLAGS = NODYNSORT };
        atomic_and_32_nv        { FLAGS = NODYNSORT };
        atomic_and_64_nv        { FLAGS = NODYNSORT };
        atomic_dec_8_nv         { FLAGS = NODYNSORT };
        atomic_dec_16_nv        { FLAGS = NODYNSORT };
        atomic_dec_32_nv        { FLAGS = NODYNSORT };
        atomic_dec_64_nv        { FLAGS = NODYNSORT };
        atomic_inc_8_nv         { FLAGS = NODYNSORT };
        atomic_inc_16_nv        { FLAGS = NODYNSORT };
        atomic_inc_32_nv        { FLAGS = NODYNSORT };
        atomic_inc_64_nv        { FLAGS = NODYNSORT };
        atomic_or_8_nv          { FLAGS = NODYNSORT };
        atomic_or_16_nv         { FLAGS = NODYNSORT };
        atomic_or_32_nv         { FLAGS = NODYNSORT };
        atomic_or_64_nv         { FLAGS = NODYNSORT };
$endif
} SUNW_1.22;

SYMBOL_VERSION SUNW_1.22 {      # SunOS 5.10 (Solaris 10)
    global:
$if _ELF32
        dladdr          { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dladdr1         { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dlclose         { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dldump          { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dlerror         { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dlinfo          { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dlmopen         { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dlopen          { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
        dlsym           { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
$elif sparcv9
        dladdr          { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dladdr1         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dlclose         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dldump          { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dlerror         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dlinfo          { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dlmopen         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dlopen          { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
        dlsym           { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
$elif amd64
        dladdr          { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dladdr1         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dlamd64getunwind { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dlclose         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dldump          { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dlerror         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dlinfo          { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dlmopen         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dlopen          { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
        dlsym           { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
$else
$error unknown platform
$endif

    protected:
        alphasort;
        _alphasort;
```



```
        atomic_add_16;
        atomic_add_16_nv;
        atomic_add_32;
        atomic_add_32_nv;
        atomic_add_64;
        atomic_add_64_nv;
        atomic_add_long         { FLAGS = NODYNSORT };
        atomic_add_long_nv      { FLAGS = NODYNSORT };
        atomic_and_32;
        atomic_and_uint         { FLAGS = NODYNSORT };
        atomic_or_32;
        atomic_or_uint          { FLAGS = NODYNSORT };
        _Exit;
        getisax;
        _getisax;
        getopt_clip;
        _getopt_clip;
        getopt_long;
        _getopt_long;
        getopt_long_only;
        _getopt_long_only;
        getpeerucred;
        _getpeerucred;
        getpflags;
        _getpflags;
        getppriv;
        _getppriv;
        getprivimplinfo;
        _getprivimplinfo;
        getzoneid;
        getzoneidbyname;
        getzonenamebyid;
        imaxabs;
        imaxdiv;
        isblank;
        iswblank;
        port_alert;
        port_associate;
        port_create;
        port_dissociate;
        port_get;
        port_getn;
        port_send;
        port_sendn;
        posix_openpt;
        posix_spawn;
        posix_spawnattr_destroy;
        posix_spawnattr_getflags;
        posix_spawnattr_getpgroup;
        posix_spawnattr_getschedparam;
        posix_spawnattr_getschedpolicy;
        posix_spawnattr_getsigdefault;
        posix_spawnattr_getsigmask;
        posix_spawnattr_init;
        posix_spawnattr_setflags;
        posix_spawnattr_setpgroup;
        posix_spawnattr_setschedparam;
        posix_spawnattr_setschedpolicy;
        posix_spawnattr_setsigdefault;
        posix_spawnattr_setsigmask;
        posix_spawn_file_actions_addclose;
        posix_spawn_file_actions_adddup2;
        posix_spawn_file_actions_addopen;
        posix_spawn_file_actions_destroy;
        posix_spawn_file_actions_init;
        posix_spawnp;
```

```
788         priv_addset;
789         _priv_addset;
790         priv_allocset;
791         _priv_allocset;
792         priv_copyset;
793         _priv_copyset;
794         priv_delset;
795         _priv_delset;
796         priv_emptyset;
797         _priv_emptyset;
798         priv_fillset;
799         _priv_fillset;
800         __priv_free_info;
801         priv_freeset;
802         _priv_freeset;
803         priv_getbyname;
804         _priv_getbyname;
805         __priv_getbyname;
806         priv_getbynum;
807         _priv_getbynum;
808         __priv_getbynum;
809         __priv_getdata;
810         priv_getsetbyname;
811         _priv_getsetbyname;
812         __priv_getsetbyname;
813         priv_getsetbynum;
814         _priv_getsetbynum;
815         __priv_getsetbynum;
816         priv_gettext;
817         _priv_gettext;
818         priv_ineffect;
819         _priv_ineffect;
820         priv_intersect;
821         _priv_intersect;
822         priv_inverse;
823         _priv_inverse;
824         priv_isemptyset;
825         _priv_isemptyset;
826         priv_isequalset;
827         _priv_isequalset;
828         priv_isfullset;
829         _priv_isfullset;
830         priv_ismember;
831         _priv_ismember;
832         priv_issubset;
833         _priv_issubset;
834         __priv_parse_info;
835         priv_set;
836         _priv_set;
837         priv_set_to_str;
838         _priv_set_to_str;
839         __priv_set_to_str;
840         priv_str_to_set;
841         _priv_str_to_set;
842         priv_union;
843         _priv_union;
844         pselect;
845         pthread_attr_getstack;
846         pthread_attr_setstack;
847         pthread_barrierattr_destroy;
848         pthread_barrierattr_getpshared;
849         pthread_barrierattr_init;
850         pthread_barrierattr_setpshared;
851         pthread_barrier_destroy;
852         pthread_barrier_init;
853         pthread_barrier_wait;
```



```
854         pthread_condattr_getclock;
855         pthread_condattr_setclock;
856         pthread_mutexattr_getrobust_np  { FLAGS = NODYNSORT };
857         pthread_mutexattr_setrobust_np  { FLAGS = NODYNSORT };
858         pthread_mutex_consistent_np     { FLAGS = NODYNSORT };
859         pthread_mutex_reltimedlock_np;
860         pthread_mutex_timedlock;
861         pthread_rwlock_reltimedrdlock_np;
862         pthread_rwlock_reltimedwrlock_np;
863         pthread_rwlock_timedrdlock;
864         pthread_rwlock_timedwrlock;
865         pthread_setschedprio;
866         pthread_spin_destroy;
867         pthread_spin_init;
868         pthread_spin_lock;
869         pthread_spin_trylock;
870         pthread_spin_unlock;
871         rctlblk_set_recipient_pid;
872         scandir;
873         _scandir;
874         schedctl_exit;
875         schedctl_init;
876         schedctl_lookup;
877         sema_reltimedwait;
878         sema_timedwait;
879         setenv;
880         setpflags;
881         _setpflags;
882         setppriv;
883         _setppriv;
884         strerror_r;
885         strtof;
886         strtoimax;
887         strtold;
888         strtoumax;
889         ucred_free;
890         _ucred_free;
891         ucred_get;
892         _ucred_get;
893         ucred_getegid;
894         _ucred_getegid;
895         ucred_geteuid;
896         _ucred_geteuid;
897         ucred_getgroups;
898         _ucred_getgroups;
899         ucred_getpflags;
900         _ucred_getpflags;
901         ucred_getpid;
902         _ucred_getpid;
903         ucred_getprivset;
904         _ucred_getprivset;
905         ucred_getprojid;
906         _ucred_getprojid;
907         ucred_getrgid;
908         _ucred_getrgid;
909         ucred_getruid;
910         _ucred_getruid;
911         ucred_getsgid;
912         _ucred_getsgid;
913         ucred_getsuid;
914         _ucred_getsuid;
915         ucred_getzoneid;
916         _ucred_getzoneid;
917         ucred_size;
918         _ucred_size;
919         unsetenv;
```

```
920         wcstof;
921         wcstoimax;
922         wcstold;
923         wcstoll;
924         wcstoull;
925         wcstoumax;

927 $if lf64
928         alphasort64;
929         _alphasort64;
930         pselect_large_fdset;
931         scandir64;
932         _scandir64;
933 $endif

935 $if _ELF64
936         walkcontext;
937 $endif

939 $if _sparc
940         # Note: atomic_add_[16,32,64]_nv are also defined above. Here, we add
941         # the NODYNSORT attribute to them. On this platform, they are aliases
942         # for the non-_nv versions. If that is changed, these lines should be
943         # removed.
944         atomic_add_16_nv        { FLAGS = NODYNSORT };
945         atomic_add_32_nv        { FLAGS = NODYNSORT };
946         atomic_add_64_nv        { FLAGS = NODYNSORT };
947 $endif

949 $if i386
950         # Note: atomic_add_64_nv is also defined above. Here, we add the
951         # NODYNSORT attribute to it. On this platform, it is an aliases for
952         # atomic_add_64. If that is changed, this line should be removed.
953         atomic_add_64_nv        { FLAGS = NODYNSORT };
954 $endif

956 $if amd64
957         # Exception unwind APIs required by the amd64 ABI
958         _SUNW_Unwind_DeleteException;
959         _SUNW_Unwind_ForcedUnwind;
960         _SUNW_Unwind_GetCFA;
961         _SUNW_Unwind_GetGR;
962         _SUNW_Unwind_GetIP;
963         _SUNW_Unwind_GetLanguageSpecificData;
964         _SUNW_Unwind_GetRegionStart;
965         _SUNW_Unwind_RaiseException;
966         _SUNW_Unwind_Resume;
967         _SUNW_Unwind_SetGR;
968         _SUNW_Unwind_SetIP;
969         _UA_CLEANUP_PHASE;
970         _UA_FORCE_UNWIND;
971         _UA_HANDLER_FRAME;
972         _UA_SEARCH_PHASE;
973         _Unwind_DeleteException;
974         _Unwind_ForcedUnwind;
975         _Unwind_GetCFA;
976         _Unwind_GetGR;
977         _Unwind_GetIP;
978         _Unwind_GetLanguageSpecificData;
979         _Unwind_GetRegionStart;
980         _Unwind_RaiseException;
981         _Unwind_Resume;
982         _Unwind_SetGR;
983         _Unwind_SetIP;
984 $endif
985 } SUNW_1.21.3;
```



```
987 SYMBOL_VERSION SUNW_1.21.3 {    # SunOS 5.9 (Solaris 9) patch additions
988     protected:
989         forkall;
990 } SUNW_1.21.2;

992 SYMBOL_VERSION SUNW_1.21.2 {    # SunOS 5.9 (Solaris 9) patch additions
993     protected:
994         getustack;
995         _getustack;
996         setustack;
997         _setustack;
998         stack_getbounds;
999         _stack_getbounds;
1000        _stack_grow;
1001        stack_inbounds;
1002        _stack_inbounds;
1003        stack_setbounds;
1004        _stack_setbounds;
1005        stack_violation;
1006        _stack_violation;

1008 $if _sparc
1009        __makecontext_v2;
1010        ___makecontext_v2;
1011 $endif
1012 } SUNW_1.21.1;

1014 SYMBOL_VERSION SUNW_1.21.1 {    # SunOS 5.9 (Solaris 9) patch additions
1015     protected:
1016         crypt_gensalt;
1017 } SUNW_1.21;

1019 SYMBOL_VERSION SUNW_1.21 {      # SunOS 5.9 (Solaris 9)
1020     protected:
1021         attropen;
1022         _attropen;
1023         bind_textdomain_codeset;
1024         closefrom;
1025         _closefrom;
1026         cond_reltimedwait;
1027         dcngettext;
1028         dngettext;
1029         fchownat;
1030         _fchownat;
1031         fdopendir;
1032         _fdopendir;
1033         fdwalk;
1034         _fdwalk;
1035         fstatat;
1036         _fstatat;
1037         futimesat;
1038         _futimesat;
1039         getcpuid;
1040         _getcpuid;
1041         gethomelgroup;
1042         _gethomelgroup          { FLAGS = NODYNSORT };
1043         getpagesizes;
1044         getrctl;
1045         _getrctl;
1046         issetugid;
1047         _issetugid;
1048         _lwp_cond_reltimedwait;
1049         meminfo;
1050         _meminfo;
1051         ngettext;
```

```
1052         openat;
1053         _openat;
1054         printstack;
1055         priocntl;
1056         priocntlset;
1057         pset_getattr;
1058         pset_getloadavg;
1059         pset_list;
1060         pset_setattr;
1061         pthread_cond_reltimedwait_np;
1062         rctlblk_get_enforced_value;
1063         rctlblk_get_firing_time;
1064         rctlblk_get_global_action;
1065         rctlblk_get_global_flags;
1066         rctlblk_get_local_action;
1067         rctlblk_get_local_flags;
1068         rctlblk_get_privilege;
1069         rctlblk_get_recipient_pid;
1070         rctlblk_get_value;
1071         rctlblk_set_local_action;
1072         rctlblk_set_local_flags;
1073         rctlblk_set_privilege;
1074         rctlblk_set_value;
1075         rctlblk_size;
1076         rctl_walk;
1077         renameat;
1078         setrctl;
1079         _setrctl;
1080         unlinkat;
1081         _unlinkat;
1082         vfscanf;
1083         _vfscanf;
1084         vfwscanf;
1085         vscanf;
1086         _vscanf;
1087         vsscanf;
1088         _vsscanf;
1089         vswscanf;
1090         vwscanf;

1092 $if _ELF32
1093         walkcontext;
1094 $endif

1096 $if lf64
1097         attropen64;
1098         _attropen64;
1099         fstatat64;
1100         _fstatat64;
1101         openat64;
1102         _openat64;
1103 $endif
1104 } SUNW_1.20.4;

1106 SYMBOL_VERSION SUNW_1.20.4 {    # SunOS 5.8 (Solaris 8) patch additions
1107     protected:
1108         semtimedop;
1109         _semtimedop;
1110 } SUNW_1.20.1;

1112 SYMBOL_VERSION SUNW_1.20.1 {    # SunOS 5.8 (Solaris 8) patch additions
1113     protected:
1114         getacct;
1115         _getacct;
1116         getprojid;
1117         _getprojid;
```



```
1118         gettaskid;
1119         _gettaskid;
1120         msgids;
1121         _msgids;
1122         msgsnap;
1123         _msgsnap;
1124         putacct;
1125         _putacct;
1126         semids;
1127         _semids;
1128         settaskid;
1129         _settaskid;
1130         shmids;
1131         _shmids;
1132         wracct;
1133         _wracct;
1134 } SUNW_1.20;

1136 SYMBOL_VERSION SUNW_1.20 {      # SunOS 5.8 (Solaris 8)
1137     protected:
1138         getextmntent;
1139         resetmnttab;
1140 } SUNW_1.19;

1142 SYMBOL_VERSION SUNW_1.19 {
1143     protected:
1144         strlcat;
1145         strlcpy;
1146         umount2;
1147         _umount2;
1148 } SUNW_1.18.1;

1150 SYMBOL_VERSION SUNW_1.18.1 {
1151     protected:
1152         __fsetlocking;
1153 } SUNW_1.18;

1155 SYMBOL_VERSION SUNW_1.18 {      # SunOS 5.7 (Solaris 7)
1156     protected:
1157         btowc;
1158         __fbufsize;
1159         __flbf;
1160         _flushlbf;
1161         __fpending;
1162         __fpurge;
1163         __freadable;
1164         __freading;
1165         fwide;
1166         fwprintf;
1167         __fwritable;
1168         __fwriting;
1169         fwscanf;
1170         getloadavg;
1171         isaexec;
1172         mbrlen;
1173         mbrtowc;
1174         mbsinit;
1175         mbsrtowcs;
1176         pcsample;
1177         pthread_attr_getguardsize;
1178         pthread_attr_setguardsize;
1179         pthread_getconcurrency;
1180         pthread_mutexattr_gettype;
1181         pthread_mutexattr_settype;
1182         pthread_rwlockattr_destroy;
1183         pthread_rwlockattr_getpshared;
```

```
1184         pthread_rwlockattr_init;
1185         pthread_rwlockattr_setpshared;
1186         pthread_rwlock_destroy;
1187         pthread_rwlock_init;
1188         pthread_rwlock_rdlock;
1189         pthread_rwlock_tryrdlock;
1190         pthread_rwlock_trywrlock;
1191         pthread_rwlock_unlock;
1192         pthread_rwlock_wrlock;
1193         pthread_setconcurrency;
1194         swprintf;
1195         swscanf;
1196         __sysconf_xpg5;
1197         vfwprintf;
1198         vswprintf;
1199         vwprintf;
1200         wcrtomb;
1201         wcsrtombs;
1202         wcsstr;
1203         wctob;
1204         wmemchr;
1205         wmemcmp;
1206         wmemcpy;
1207         wmemmove;
1208         wmemset;
1209         wprintf;
1210         wscanf;

1212 $if _ELF32
1213         select_large_fdset;
1214 $endif
1215 } SUNW_1.17;

1217 # The empty versions SUNW_1.2 through SUNW_1.17 must be preserved because
1218 # applications built on Solaris 2.6 Beta (when they did contain symbols)
1219 # may depend on them.  All symbol content for SunOS 5.6 is now in SUNW_1.1

1221 SYMBOL_VERSION SUNW_1.17 {
1222     protected:
1223         SUNW_1.17;
1224 } SUNW_1.16;

1226 SYMBOL_VERSION SUNW_1.16 {
1227     protected:
1228         SUNW_1.16;
1229 } SUNW_1.15;

1231 SYMBOL_VERSION SUNW_1.15 {
1232     protected:
1233         SUNW_1.15;
1234 } SUNW_1.14;

1236 SYMBOL_VERSION SUNW_1.14 {
1237     protected:
1238         SUNW_1.14;
1239 } SUNW_1.13;

1241 SYMBOL_VERSION SUNW_1.13 {
1242     protected:
1243         SUNW_1.13;
1244 } SUNW_1.12;

1246 SYMBOL_VERSION SUNW_1.12 {
1247     protected:
1248         SUNW_1.12;
1249 } SUNW_1.11;
```



```
1251 SYMBOL_VERSION SUNW_1.11 {
1252     protected:
1253         SUNW_1.11;
1254 } SUNW_1.10;

1256 SYMBOL_VERSION SUNW_1.10 {
1257     protected:
1258         SUNW_1.10;
1259 } SUNW_1.9;

1261 SYMBOL_VERSION SUNW_1.9 {
1262     protected:
1263         SUNW_1.9;
1264 } SUNW_1.8;

1266 SYMBOL_VERSION SUNW_1.8 {
1267     protected:
1268         SUNW_1.8;
1269 } SUNW_1.7;

1271 SYMBOL_VERSION SUNW_1.7 {
1272     protected:
1273         SUNW_1.7;
1274 } SUNW_1.6;

1276 SYMBOL_VERSION SUNW_1.6 {
1277     protected:
1278         SUNW_1.6;
1279 } SUNW_1.5;

1281 SYMBOL_VERSION SUNW_1.5 {
1282     protected:
1283         SUNW_1.5;
1284 } SUNW_1.4;

1286 SYMBOL_VERSION SUNW_1.4 {
1287     protected:
1288         SUNW_1.4;
1289 } SUNW_1.3;

1291 SYMBOL_VERSION SUNW_1.3 {
1292     protected:
1293         SUNW_1.3;
1294 } SUNW_1.2;

1296 SYMBOL_VERSION SUNW_1.2 {
1297     protected:
1298         SUNW_1.2;
1299 } SUNW_1.1;

1301 SYMBOL_VERSION SUNW_1.1 {       # SunOS 5.6 (Solaris 2.6)
1302     global:
1303         __loc1;
1304     protected:
1305         basename;
1306         bindtextdomain;
1307         bsd_signal;
1308         dbm_clearerr;
1309         dbm_error;
1310         dcgettext;
1311         dgettext;
1312         directio;
1313         dirname;
1314         endusershell;
1315         _exithandle;
```

```
1316        fgetwc;
1317        fgetws;
1318        fpgetround;
1319        fpsetround;
1320        fputwc;
1321        fputws;
1322        fseeko;
1323        ftello;
1324        ftrylockfile;
1325        getexecname;
1326        _getexecname;
1327        getpassphrase;
1328        gettext;
1329        getusershell;
1330        getwc;
1331        getwchar;
1332        getws;
1333        isenglish;
1334        isideogram;
1335        isnumber;
1336        isphonogram;
1337        isspecial;
1338        iswalnum;
1339        iswalpha;
1340        iswcntrl;
1341        iswctype;
1342        iswdigit;
1343        iswgraph;
1344        iswlower;
1345        iswprint;
1346        iswpunct;
1347        iswspace;
1348        iswupper;
1349        iswxdigit;
1350        ___loc1;
1351        _longjmp;
1352        _lwp_sema_trywait;
1353        ntp_adjtime;
1354        _ntp_adjtime;
1355        ntp_gettime;
1356        _ntp_gettime;
1357        __posix_asctime_r;
1358        __posix_ctime_r;
1359        __posix_getgrgid_r;
1360        __posix_getgrnam_r;
1361        __posix_getlogin_r;
1362        __posix_getpwnam_r;
1363        __posix_getpwuid_r;
1364        __posix_sigwait;
1365        __posix_ttyname_r;
1366        pset_assign;
1367        pset_bind;
1368        pset_create;
1369        pset_destroy;
1370        pset_info;
1371        pthread_atfork;
1372        pthread_attr_destroy;
1373        pthread_attr_getdetachstate;
1374        pthread_attr_getinheritsched;
1375        pthread_attr_getschedparam;
1376        pthread_attr_getschedpolicy;
1377        pthread_attr_getscope;
1378        pthread_attr_getstackaddr;
1379        pthread_attr_getstacksize;
1380        pthread_attr_init;
1381        pthread_attr_setdetachstate;
```



```
1382        pthread_attr_setinheritsched;
1383        pthread_attr_setschedparam;
1384        pthread_attr_setschedpolicy;
1385        pthread_attr_setscope;
1386        pthread_attr_setstackaddr;
1387        pthread_attr_setstacksize;
1388        pthread_cancel;
1389        __pthread_cleanup_pop;
1390        __pthread_cleanup_push;
1391        pthread_create;
1392        pthread_detach;
1393        pthread_equal;
1394        pthread_exit;
1395        pthread_getschedparam;
1396        pthread_getspecific;
1397        pthread_join;
1398        pthread_key_create;
1399        pthread_key_delete;
1400        pthread_kill;
1401        pthread_once;
1402        pthread_self;
1403        pthread_setcancelstate;
1404        pthread_setcanceltype;
1405        pthread_setschedparam;
1406        pthread_setspecific;
1407        pthread_sigmask;
1408        pthread_testcancel;
1409        putwc;
1410        putwchar;
1411        putws;
1412        regcmp;
1413        regex;
1414        resolvepath;
1415        _resolvepath;
1416        rwlock_destroy          { FLAGS = NODYNSORT };
1417        _rwlock_destroy         { FLAGS = NODYNSORT };
1418        sema_destroy;
1419        _sema_destroy;
1420        _setjmp;
1421        setusershell;
1422        siginterrupt;
1423        sigstack;
1424        snprintf;
1425        strtows;
1426        sync_instruction_memory;
1427        textdomain;
1428        thr_main;
1429        towctrans;
1430        towlower;
1431        towupper;
1432        ungetwc;
1433        vsnprintf;
1434        watoll;
1435        wcscat;
1436        wcschr;
1437        wcscmp;
1438        wcscoll;
1439        wcscpy;
1440        wcscspn;
1441        wcsftime;
1442        wcslen;
1443        wcsncat;
1444        wcsncmp;
1445        wcsncpy;
1446        wcspbrk;
1447        wcsrchr;
```

```
1448         wcsspn;
1449         wcstod;
1450         wcstok;
1451         wcstol;
1452         wcstoul;
1453         wcswcs;
1454         wcswidth;
1455         wcsxfrm;
1456         wctrans;
1457         wctype;
1458         wcwidth;
1459         wscasecmp;
1460         wscat;
1461         wschr;
1462         wscmp;
1463         wscol;
1464         wscoll;
1465         wscpy;
1466         wscspn;
1467         wsdup;
1468         wslen;
1469         wsncasecmp;
1470         wsncat;
1471         wsncmp;
1472         wsncpy;
1473         wspbrk;
1474         wsprintf;
1475         wsrchr;
1476         wsscanf;
1477         wsspn;
1478         wstod;
1479         wstok;
1480         wstol;
1481         wstoll;
1482         wstostr;
1483         wsxfrm;
1484         __xpg4_putmsg;
1485         __xpg4_putpmsg;

1487 $if lf64
1488         creat64;
1489         _creat64;
1490         fgetpos64;
1491         fopen64;
1492         freopen64;
1493         fseeko64;
1494         fsetpos64;
1495         fstat64;
1496         _fstat64;
1497         fstatvfs64;
1498         _fstatvfs64;
1499         ftello64;
1500         ftruncate64;
1501         _ftruncate64;
1502         ftw64;
1503         _ftw64;
1504         getdents64;
1505         _getdents64;
1506         getrlimit64;
1507         _getrlimit64;
1508         lockf64;
1509         _lockf64;
1510         lseek64;
1511         _lseek64;
1512         lstat64;
1513         _lstat64;
```



```
1514         mkstemp64;
1515         _mkstemp64;
1516         mmap64;
1517         _mmap64;
1518         nftw64;
1519         _nftw64;
1520         open64;
1521         _open64;
1522         __posix_readdir_r;
1523         pread64;
1524         _pread64;
1525         pwrite64;
1526         _pwrite64;
1527         readdir64;
1528         _readdir64;
1529         readdir64_r;
1530         _readdir64_r;
1531         setrlimit64;
1532         _setrlimit64;
1533         s_fcntl;
1534         _s_fcntl                { FLAGS = NODYNSORT };
1535         s_ioctl;
1536         stat64;
1537         _stat64;
1538         statvfs64;
1539         _statvfs64;
1540         tell64;
1541         _tell64;
1542         tmpfile64;
1543         truncate64;
1544         _truncate64;
1545         _xftw64;
1546 $endif

1548 $if _sparc
1549         __flt_rounds;
1550 $endif
1551 } SUNW_0.9;

1553 SYMBOL_VERSION SUNW_0.9 {       # SunOS 5.5 (Solaris 2.5)
1554     protected:
1555         acl;
1556         bcmp;
1557         bcopy;
1558         bzero;
1559         facl;
1560         ftime;
1561         getdtablesize;
1562         gethostid;
1563         gethostname;
1564         getpagesize;
1565         getpriority;
1566         getrusage;
1567         getwd;
1568         index;
1569         initstate;
1570         killpg;
1571         _nsc_trydoorcall;
1572         pthread_condattr_destroy;
1573         pthread_condattr_getpshared;
1574         pthread_condattr_init;
1575         pthread_condattr_setpshared;
1576         pthread_cond_broadcast;
1577         pthread_cond_destroy;
1578         pthread_cond_init;
1579         pthread_cond_signal;
```

```
1580         pthread_cond_timedwait;
1581         pthread_cond_wait;
1582         pthread_mutexattr_destroy;
1583         pthread_mutexattr_getprioceiling;
1584         pthread_mutexattr_getprotocol;
1585         pthread_mutexattr_getpshared;
1586         pthread_mutexattr_init;
1587         pthread_mutexattr_setprioceiling;
1588         pthread_mutexattr_setprotocol;
1589         pthread_mutexattr_setpshared;
1590         pthread_mutex_destroy;
1591         pthread_mutex_getprioceiling;
1592         pthread_mutex_init;
1593         pthread_mutex_lock;
1594         pthread_mutex_setprioceiling;
1595         pthread_mutex_trylock;
1596         pthread_mutex_unlock;
1597         random;
1598         reboot;
1599         re_comp;
1600         re_exec;
1601         rindex;
1602         setbuffer;
1603         sethostname;
1604         setlinebuf;
1605         setpriority;
1606         setregid;
1607         setreuid;
1608         setstate;
1609         srandom;
1610         thr_min_stack;
1611         thr_stksegment;
1612         ualarm;
1613         usleep;
1614         wait3;
1615         wait4;
1616 } SUNW_0.8;

1618 SYMBOL_VERSION SUNW_0.8 {       # SunOS 5.4 (Solaris 2.4)
1619     global:
1620         __xpg4                  { FLAGS = NODIRECT };
1621     protected:
1622         addsev;
1623         cond_broadcast          { FLAGS = NODYNSORT };
1624         cond_destroy            { FLAGS = NODYNSORT };
1625         cond_init;
1626         cond_signal             { FLAGS = NODYNSORT };
1627         cond_timedwait;
1628         cond_wait;
1629         confstr;
1630         fnmatch;
1631         _getdate_err_addr;
1632         glob;
1633         globfree;
1634         iconv;
1635         iconv_close;
1636         iconv_open;
1637         lfmt;
1638         mutex_destroy           { FLAGS = NODYNSORT };
1639         mutex_init;
1640         mutex_lock              { FLAGS = NODYNSORT };
1641         mutex_trylock           { FLAGS = NODYNSORT };
1642         mutex_unlock            { FLAGS = NODYNSORT };
1643         pfmt;
1644         regcomp;
1645         regerror;
```



```
1646         regexec;
1647         regfree;
1648         rwlock_init;
1649         rw_rdlock               { FLAGS = NODYNSORT };
1650         rw_read_held;
1651         rw_tryrdlock            { FLAGS = NODYNSORT };
1652         rw_trywrlock            { FLAGS = NODYNSORT };
1653         rw_unlock               { FLAGS = NODYNSORT };
1654         rw_write_held;
1655         rw_wrlock               { FLAGS = NODYNSORT };
1656         sema_held;
1657         sema_init;
1658         sema_post;
1659         sema_trywait;
1660         sema_wait;
1661         setcat;
1662         sigfpe;
1663         strfmon;
1664         strptime;
1665         thr_continue;
1666         thr_create;
1667         thr_exit;
1668         thr_getconcurrency;
1669         thr_getprio;
1670         thr_getspecific;
1671         thr_join;
1672         thr_keycreate;
1673         thr_kill;
1674         thr_self                { FLAGS = NODYNSORT };
1675         thr_setconcurrency;
1676         thr_setprio;
1677         thr_setspecific;
1678         thr_sigsetmask;
1679         thr_suspend;
1680         thr_yield;
1681         vlfmt;
1682         vpfmt;
1683         wordexp;
1684         wordfree;
1685 } SUNW_0.7;

1687 SYMBOL_VERSION SUNW_0.7 {       # SunOS 5.3 (Solaris 2.3)
1688     global:
1689         altzone;
1690         _ctype;
1691         isnanf                  { TYPE = FUNCTION; FILTER = libm.so.2 };
1692         lone;
1693         lten;
1694         lzero;
1695         memalign                { FLAGS = NODIRECT };
1696         modff                   { TYPE = FUNCTION; FILTER = libm.so.2 };
1697         nss_default_finders;
1698         _sibuf;
1699         _sobuf;
1700         _sys_buslist;
1701         _sys_cldlist;
1702         _sys_fpelist;
1703         _sys_illlist;
1704         _sys_segvlist;
1705         _sys_siginfolistp;
1706         _sys_siglist;
1707         _sys_siglistn;
1708         _sys_siglistp;
1709         _sys_traplist;
1710         valloc                  { FLAGS = NODIRECT };
```

```
$if _ELF32
        _bufendtab;
        _lastbuf;
        sys_errlist;
        sys_nerr;
        _sys_nsig;
$endif

    protected:
        a64l;
        adjtime;
        ascftime;
        _assert;
        atoll;
        brk;
        __builtin_alloca;
        cftime;
        closelog;
        csetcol;
        csetlen;
        ctermid_r;
        dbm_close;
        dbm_delete;
        dbm_fetch;
        dbm_firstkey;
        dbm_nextkey;
        dbm_open;
        dbm_store;
        decimal_to_double;
        decimal_to_extended;
        decimal_to_quadruple;
        decimal_to_single;
        double_to_decimal;
        drand48;
        econvert;
        ecvt;
        endnetgrent;
        endspent;
        endutent;
        endutxent;
        erand48;
        euccol;
        euclen;
        eucscol;
        extended_to_decimal;
        fchroot;
        fconvert;
        fcvt;
        ffs;
        fgetspent;
        fgetspent_r;
        _filbuf;
        file_to_decimal;
        finite;
        _flsbuf;
        fork1                   { FLAGS = NODYNSORT };
        fpclass;
        fpgetmask;
        fpgetsticky;
        fpsetmask;
        fpsetsticky;
        fstatfs;
        ftruncate;
        ftw;
        func_to_decimal;
        gconvert;
```



```
        gcvt;
        getdents;
        gethrtime;
        gethrvtime;
        getmntany;
        getmntent;
        getnetgrent;
        getnetgrent_r;
        getpw;
        getspent;
        getspent_r;
        getspnam;
        getspnam_r;
        getutent;
        getutid;
        getutline;
        getutmp;
        getutmpx;
        getutxent;
        getutxid;
        getutxline;
        getvfsany;
        getvfsent;
        getvfsfile;
        getvfsspec;
        getwidth;
        gsignal;
        hasmntopt;
        innetgr;
        insque;
        _insque;
        jrand48;
        l64a;
        ladd;
        lckpwdf;
        lcong48;
        ldivide;
        lexp10;
        llabs;
        lldiv;
        llog10;
        llseek;
        lltostr;
        lmul;
        lrand48;
        lshiftl;
        lsub;
        _lwp_cond_broadcast;
        _lwp_cond_signal;
        _lwp_cond_timedwait;
        _lwp_cond_wait;
        _lwp_continue;
        _lwp_info;
        _lwp_kill;
        _lwp_mutex_lock;
        _lwp_mutex_trylock;
        _lwp_mutex_unlock;
        _lwp_self;
        _lwp_sema_init;
        _lwp_sema_post;
        _lwp_sema_wait;
        _lwp_suspend;
        madvise;
        __major;
        __makedev;
        mincore;
```

```
1844         __minor;
1845         mkstemp;
1846         _mkstemp;
1847         mlockall;
1848         mrand48;
1849         munlockall;
1850         _mutex_held             { FLAGS = NODYNSORT };
1851         _mutex_lock             { FLAGS = NODYNSORT };
1852         nrand48;
1853         _nss_netdb_aliases;
1854         _nss_XbyY_buf_alloc;
1855         _nss_XbyY_buf_free;
1856         __nsw_extended_action;
1857         __nsw_freeconfig;
1858         __nsw_getconfig;
1859         openlog;
1860         plock;
1861         p_online;
1862         pread;
1863         __priocntl;
1864         __priocntlset;
1865         processor_bind;
1866         processor_info;
1867         psiginfo;
1868         psignal;
1869         putpwent;
1870         putspent;
1871         pututline;
1872         pututxline;
1873         pwrite;
1874         qeconvert;
1875         qecvt;
1876         qfconvert;
1877         qfcvt;
1878         qgconvert;
1879         qgcvt;
1880         quadruple_to_decimal;
1881         realpath;
1882         remque;
1883         _remque;
1884         _rw_read_held;
1885         _rw_write_held;
1886         seconvert;
1887         seed48;
1888         select;
1889         _sema_held;
1890         setegid;
1891         seteuid;
1892         setlogmask;
1893         setnetgrent;
1894         setspent;
1895         settimeofday;
1896         setutent;
1897         setutxent;
1898         sfconvert;
1899         sgconvert;
1900         sig2str;
1901         sigwait;
1902         single_to_decimal;
1903         srand48;
1904         ssignal;
1905         statfs;
1906         str2sig;
1907         strcasecmp;
1908         string_to_decimal;
1909         strncasecmp;
```



```
1910         strsignal;
1911         strtoll;
1912         strtoull;
1913         swapctl;
1914         _syscall;
1915         sysfs;
1916         syslog;
1917         _syslog;
1918         tmpnam_r;
1919         truncate;
1920         ttyslot;
1921         uadmin;
1922         ulckpwdf;
1923         ulltostr;
1924         unordered;
1925         updwtmp;
1926         updwtmpx;
1927         ustat;
1928         utimes;
1929         utmpname;
1930         utmpxname;
1931         vfork;
1932         vhangup;
1933         vsyslog;
1934         yield;

1936 $if i386
1937         # Note: _syscall is also defined above. Here, we add the NODYNSORT
1938         # attribute to it. On this platform, it is an alias to syscall.
1939         # If that is changed, this lines should be removed.
1940         _syscall                { FLAGS = NODYNSORT };
1941 $endif

1943 # The 32-bit sparc ABI requires SISCD_2.3. On other platforms, those symbols
1944 # go directly into SUNW_0.7.
1945 $if sparc32
1946 } SISCD_2.3;

1948 SYMBOL_VERSION SISCD_2.3 {
1949 $endif

1951     global:
1952         errno                   { FLAGS = NODIRECT };
1953         _iob;

1955     protected:
1956         addseverity;
1957         _addseverity;
1958         asctime_r;
1959         crypt;
1960         _crypt;
1961         ctime_r;
1962         encrypt;
1963         _encrypt;
1964         endgrent;
1965         endpwent;
1966         ___errno;
1967         fgetgrent;
1968         fgetgrent_r;
1969         fgetpwent;
1970         fgetpwent_r;
1971         flockfile;
1972         funlockfile;
1973         getchar_unlocked;
1974         getc_unlocked;
1975         getgrent;
```

```
1976         getgrent_r;
1977         getgrgid_r;
1978         getgrnam_r;
1979         getitimer;
1980         _getitimer;
1981         getlogin_r;
1982         getpwent;
1983         getpwent_r;
1984         getpwnam_r;
1985         getpwuid_r;
1986         gettimeofday;
1987         _gettimeofday;
1988         gmtime_r;
1989         localtime_r;
1990         putchar_unlocked;
1991         putc_unlocked;
1992         rand_r;
1993         readdir_r;
1994         setgrent;
1995         setitimer;
1996         _setitimer;
1997         setkey;
1998         _setkey;
1999         setpwent;
2000         strtok_r;
2001         sysinfo;
2002         _sysinfo;
2003         ttyname_r;

2005 $if _ELF32
2006         __div64;
2007         __mul64;
2008         __rem64;
2009         __udiv64;
2010         __urem64;
2011 $endif

2013 $if sparc32
2014         __dtoll;
2015         __dtoull;
2016         __ftoll;
2017         __ftoull;
2018         _Q_lltoq;
2019         _Q_qtoll;
2020         _Q_qtoull;
2021         _Q_ulltoq;
2022         sbrk;
2023         _sbrk;
2024         __umul64                { FLAGS = NODYNSORT };  # Same address as __mul6
2025 $endif

2027 # On 32-bit platforms, the following symbols go into SYSVABI_1.3, but on
2028 # other platforms they go directly into the current version (which will be
2029 # either SUNW_0.7, or SISCD_2.3, depending on the similar issue described above.
2030 $if _ELF32
2031 } SYSVABI_1.3;

2033 SYMBOL_VERSION SYSVABI_1.3 {
2034 $endif

2036     global:
2037         _altzone;
2038         calloc                  { FLAGS = NODIRECT };
2039         __ctype;
2040         daylight;
2041         _daylight;
```



```
2042         environ                 { FLAGS = NODIRECT };
2043         _environ                { FLAGS = NODIRECT };
2044         free                    { FLAGS = NODIRECT };
2045         frexp                   { TYPE = FUNCTION; FILTER = libm.so.2 };
2046         getdate_err;
2047         _getdate_err;
2048         getenv;
2049         __huge_val;
2050         __iob;
2051         isnan                   { TYPE = FUNCTION; FILTER = libm.so.2 };
2052         _isnan                  { TYPE = FUNCTION; FILTER = libm.so.2 };
2053         isnand                  { TYPE = FUNCTION; FILTER = libm.so.2 };
2054         _isnand                 { TYPE = FUNCTION; FILTER = libm.so.2 };
2055         ldexp                   { TYPE = FUNCTION; FILTER = libm.so.2 };
2056         logb                    { TYPE = FUNCTION; FILTER = libm.so.2 };
2057         malloc                  { FLAGS = NODIRECT };
2058         memcmp;
2059         memcpy;
2060         memmove;
2061         memset;
2062         modf                    { TYPE = FUNCTION; FILTER = libm.so.2 };
2063         _modf                   { TYPE = FUNCTION; FILTER = libm.so.2 };
2064         nextafter               { TYPE = FUNCTION; FILTER = libm.so.2 };
2065         _nextafter              { TYPE = FUNCTION; FILTER = libm.so.2 };
2066         _numeric;
2067         optarg;
2068         opterr;
2069         optind;
2070         optopt;
2071         realloc                 { FLAGS = NODIRECT };
2072         scalb                   { TYPE = FUNCTION; FILTER = libm.so.2 };
2073         _scalb                  { TYPE = FUNCTION; FILTER = libm.so.2 };
2074         timezone;
2075         _timezone;
2076         tzname;
2077         _tzname;
2078 $if i386
2079         _fp_hw;
2080 $endif

2082     protected:
2083         abort;
2084         abs;
2085         access;
2086         _access;
2087         acct;
2088         _acct;
2089         alarm;
2090         _alarm;
2091         asctime;
2092         __assert;
2093         atexit;
2094         atof;
2095         atoi;
2096         atol;
2097         bsearch;
2098         catclose;
2099         _catclose;
2100         catgets;
2101         _catgets;
2102         catopen;
2103         _catopen;
2104         cfgetispeed;
2105         _cfgetispeed;
2106         cfgetospeed;
2107         _cfgetospeed;
```

```
2108         cfsetispeed;
2109         _cfsetispeed;
2110         cfsetospeed;
2111         _cfsetospeed;
2112         chdir;
2113         _chdir;
2114         chmod;
2115         _chmod;
2116         chown;
2117         _chown;
2118         chroot;
2119         _chroot;
2120         _cleanup;
2121         clearerr;
2122         clock;
2123         _close;
2124         close;
2125         closedir;
2126         _closedir;
2127         creat;
2128         _creat;
2129         ctermid;
2130         ctime;
2131         cuserid;
2132         _cuserid;
2133         difftime;
2134         div;
2135         dup;
2136         _dup;
2137         dup2;
2138         _dup2;
2139         execl;
2140         _execl;
2141         execle;
2142         _execle;
2143         execlp;
2144         _execlp;
2145         execv;
2146         _execv;
2147         execve;
2148         _execve;
2149         execvp;
2150         _execvp;
2151         exit;
2152         _exit;
2153         fattach;
2154         _fattach;
2155         fchdir;
2156         _fchdir;
2157         fchmod;
2158         _fchmod;
2159         fchown;
2160         _fchown;
2161         fclose;
2162         fcntl;
2163         _fcntl;
2164         fdetach;
2165         _fdetach;
2166         fdopen;
2167         _fdopen;
2168         feof;
2169         ferror;
2170         fflush;
2171         fgetc;
2172         fgetpos;
2173         fgets;
```



```
2174         __filbuf;
2175         fileno;
2176         _fileno;
2177         __flsbuf;
2178         fmtmsg;
2179         _fmtmsg;
2180         fopen;
2181         _fork;
2182         fork;
2183         fpathconf;
2184         _fpathconf;
2185         fprintf;
2186         fputc;
2187         fputs;
2188         fread;
2189         freopen;
2190         fscanf;
2191         fseek;
2192         fsetpos;
2193         fstat;
2194         _fstat;
2195         fstatvfs;
2196         _fstatvfs;
2197         fsync;
2198         _fsync;
2199         ftell;
2200         ftok;
2201         _ftok;
2202         fwrite;
2203         getc;
2204         getchar;
2205         getcontext;
2206         _getcontext;
2207         getcwd;
2208         _getcwd;
2209         getdate;
2210         _getdate;
2211         getegid;
2212         _getegid;
2213         geteuid;
2214         _geteuid;
2215         getgid;
2216         _getgid;
2217         getgrgid;
2218         getgrnam;
2219         getgroups;
2220         _getgroups;
2221         getlogin;
2222         getmsg;
2223         _getmsg;
2224         getopt;
2225         _getopt;
2226         getpass;
2227         _getpass;
2228         getpgid;
2229         _getpgid;
2230         getpgrp;
2231         _getpgrp;
2232         getpid;
2233         _getpid;
2234         getpmsg;
2235         _getpmsg;
2236         getppid;
2237         _getppid;
2238         getpwnam;
2239         getpwuid;
```

```
2240         getrlimit;
2241         _getrlimit;
2242         gets;
2243         getsid;
2244         _getsid;
2245         getsubopt;
2246         _getsubopt;
2247         gettxt;
2248         _gettxt;
2249         getuid;
2250         _getuid;
2251         getw;
2252         _getw;
2253         gmtime;
2254         grantpt;
2255         _grantpt;
2256         hcreate;
2257         _hcreate;
2258         hdestroy;
2259         _hdestroy;
2260         hsearch;
2261         _hsearch;
2262         initgroups;
2263         _initgroups;
2264         ioctl;
2265         _ioctl;
2266         isalnum;
2267         isalpha;
2268         isascii;
2269         _isascii;
2270         isastream;
2271         _isastream;
2272         isatty;
2273         _isatty;
2274         iscntrl;
2275         isdigit;
2276         isgraph;
2277         islower;
2278         isprint;
2279         ispunct;
2280         isspace;
2281         isupper;
2282         isxdigit;
2283         kill;
2284         _kill;
2285         labs;
2286         lchown;
2287         _lchown;
2288         ldiv;
2289         lfind;
2290         _lfind;
2291         link;
2292         _link;
2293         localeconv;
2294         localtime;
2295         lockf;
2296         _lockf;
2297         longjmp;
2298         lsearch;
2299         _lsearch;
2300         lseek;
2301         _lseek;
2302         lstat;
2303         _lstat;
2304         makecontext;
2305         _makecontext;
```



```
2306         mblen;
2307         mbstowcs;
2308         mbtowc;
2309         memccpy;
2310         _memccpy;
2311         memchr;
2312         memcntl;
2313         _memcntl;
2314         mkdir;
2315         _mkdir;
2316         mkfifo;
2317         _mkfifo;
2318         mknod;
2319         _mknod;
2320         mktemp;
2321         _mktemp;
2322         mktime;
2323         mlock;
2324         _mlock;
2325         mmap;
2326         _mmap;
2327         monitor;
2328         _monitor;
2329         mount;
2330         _mount;
2331         mprotect;
2332         _mprotect;
2333         msgctl;
2334         _msgctl;
2335         msgget;
2336         _msgget;
2337         msgrcv;
2338         _msgrcv;
2339         msgsnd;
2340         _msgsnd;
2341         msync;
2342         _msync;
2343         munlock;
2344         _munlock;
2345         munmap;
2346         _munmap;
2347         nftw;
2348         _nftw;
2349         nice;
2350         _nice;
2351         nl_langinfo;
2352         _nl_langinfo;
2353         open;
2354         _open;
2355         opendir;
2356         _opendir;
2357         pathconf;
2358         _pathconf;
2359         pause;
2360         _pause;
2361         pclose;
2362         _pclose;
2363         perror;
2364         pipe;
2365         _pipe;
2366         poll;
2367         _poll;
2368         popen;
2369         _popen;
2370         printf;
2371         profil;
```

```
2372        _profil;
2373        ptsname;
2374        _ptsname;
2375        putc;
2376        putchar;
2377        putenv;
2378        _putenv;
2379        putmsg;
2380        _putmsg;
2381        putpmsg;
2382        _putpmsg;
2383        puts;
2384        putw;
2385        _putw;
2386        qsort;
2387        raise;
2388        rand;
2389        read;
2390        _read;
2391        readdir;
2392        _readdir;
2393        readlink;
2394        _readlink;
2395        readv;
2396        _readv;
2397        remove;
2398        rename;
2399        _rename;
2400        rewind;
2401        rewinddir;
2402        _rewinddir;
2403        rmdir;
2404        _rmdir;
2405        scanf;
2406        seekdir;
2407        _seekdir;
2408        semctl;
2409        _semctl;
2410        semget;
2411        _semget;
2412        semop;
2413        _semop;
2414        setbuf;
2415        setcontext;
2416        _setcontext             { FLAGS = NODYNSORT };
2417        setgid;
2418        _setgid;
2419        setgroups;
2420        _setgroups;
2421        setjmp;
2422        setlabel;
2423        setlocale;
2424        setpgid;
2425        _setpgid;
2426        setpgrp;
2427        _setpgrp;
2428        setrlimit;
2429        _setrlimit;
2430        setsid;
2431        _setsid;
2432        setuid;
2433        _setuid;
2434        setvbuf;
2435        shmat;
2436        _shmat;
2437        shmctl;
```



```
2438        _shmctl;
2439        shmdt;
2440        _shmdt;
2441        shmget;
2442        _shmget;
2443        sigaction;
2444        _sigaction              { FLAGS = NODYNSORT };
2445        sigaddset;
2446        _sigaddset;
2447        sigaltstack;
2448        _sigaltstack;
2449        sigdelset;
2450        _sigdelset;
2451        sigemptyset;
2452        _sigemptyset;
2453        sigfillset;
2454        _sigfillset;
2455        sighold;
2456        _sighold;
2457        sigignore;
2458        _sigignore;
2459        sigismember;
2460        _sigismember;
2461        siglongjmp;
2462        _siglongjmp;
2463        signal;
2464        sigpause;
2465        _sigpause;
2466        sigpending;
2467        _sigpending;
2468        sigprocmask;
2469        _sigprocmask;
2470        sigrelse;
2471        _sigrelse;
2472        sigsend;
2473        _sigsend;
2474        sigsendset;
2475        _sigsendset;
2476        sigset;
2477        _sigset;
2478        sigsetjmp;
2479        _sigsetjmp              { FLAGS = NODYNSORT };
2480        sigsuspend;
2481        _sigsuspend;
2482        sleep;
2483        _sleep;
2484        sprintf;
2485        srand;
2486        sscanf;
2487        stat;
2488        _stat;
2489        statvfs;
2490        _statvfs;
2491        stime;
2492        _stime;
2493        strcat;
2494        strchr;
2495        strcmp;
2496        strcoll;
2497        strcpy;
2498        strcspn;
2499        strdup;
2500        _strdup;
2501        strerror;
2502        strftime;
2503        strlen;
```

```
2504         strncat;
2505         strncmp;
2506         strncpy;
2507         strpbrk;
2508         strrchr;
2509         strspn;
2510         strstr;
2511         strtod;
2512         strtok;
2513         strtol;
2514         strtoul;
2515         strxfrm;
2516         swab;
2517         _swab;
2518         swapcontext;
2519         _swapcontext;
2520         symlink;
2521         _symlink;
2522         sync;
2523         _sync;
2524         sysconf;
2525         _sysconf;
2526         system;
2527         tcdrain;
2528         _tcdrain;
2529         tcflow;
2530         _tcflow;
2531         tcflush;
2532         _tcflush;
2533         tcgetattr;
2534         _tcgetattr;
2535         tcgetpgrp;
2536         _tcgetpgrp;
2537         tcgetsid;
2538         _tcgetsid;
2539         tcsendbreak;
2540         _tcsendbreak;
2541         tcsetattr;
2542         _tcsetattr;
2543         tcsetpgrp;
2544         _tcsetpgrp;
2545         tdelete;
2546         _tdelete;
2547         tell;
2548         _tell;
2549         telldir;
2550         _telldir;
2551         tempnam;
2552         _tempnam;
2553         tfind;
2554         _tfind;
2555         time;
2556         _time;
2557         times;
2558         _times;
2559         tmpfile;
2560         tmpnam;
2561         toascii;
2562         _toascii;
2563         tolower;
2564         _tolower;
2565         toupper;
2566         _toupper;
2567         tsearch;
2568         _tsearch;
2569         ttyname;
```



```
2570         twalk;
2571         _twalk;
2572         tzset;
2573         _tzset;
2574         ulimit;
2575         _ulimit;
2576         umask;
2577         _umask;
2578         umount;
2579         _umount;
2580         uname;
2581         _uname;
2582         ungetc;
2583         unlink;
2584         _unlink;
2585         unlockpt;
2586         _unlockpt;
2587         utime;
2588         _utime;
2589         vfprintf;
2590         vprintf;
2591         vsprintf;
2592         wait;
2593         _wait;
2594         waitid;
2595         _waitid;
2596         waitpid;
2597         _waitpid;
2598         wcstombs;
2599         wctomb;
2600         write;
2601         _write;
2602         writev;
2603         _writev;
2604         _xftw;

2606 $if _ELF32
2607         ptrace;
2608         _ptrace;
2609 $endif

2611 $if i386
2612         _fxstat;
2613         _lxstat;
2614         nuname;
2615         _nuname;
2616         _xmknod;
2617         _xstat;
2618 $endif

2620 $if !sparc32
2621         sbrk;
2622 $endif

2624 $if _sparc
2625         __dtou;
2626         __ftou;
2627 $endif

2629 $if sparc32
2630         .div;
2631         .mul;
2632         .rem;
2633         .stret1;
2634         .stret2;
2635         .stret4;
```

```
2636         # .stret4 and .stret8 are the same thing
2637         .stret8                 { FLAGS = NODYNSORT };
2638         .udiv;
2639         .umul;
2640         .urem;
2641         _Q_add;
2642         _Q_cmp;
2643         _Q_cmpe;
2644         _Q_div;
2645         _Q_dtoq;
2646         _Q_feq;
2647         _Q_fge;
2648         _Q_fgt;
2649         _Q_fle;
2650         _Q_flt;
2651         _Q_fne;
2652         _Q_itoq;
2653         _Q_mul;
2654         _Q_neg;
2655         _Q_qtod;
2656         _Q_qtoi;
2657         _Q_qtos;
2658         _Q_qtou;
2659         _Q_sqrt;
2660         _Q_stoq;
2661         _Q_sub;
2662         _Q_utoq;
2663 $endif

2665 $if sparcv9
2666         # __align_cpy_1 is an alias for memcpy. Filter it out of
2667         # the .SUNW_dynsymsort section
2668         __align_cpy_1           { FLAGS = NODYNSORT };
2669         __align_cpy_16;
2670         __align_cpy_2;
2671         __align_cpy_4;
2672         # __align_cpy_8 is same as __align_cpy_16
2673         __align_cpy_8           { FLAGS = NODYNSORT };
2674         __dtoul;
2675         __ftoul;
2676         _Qp_add;
2677         _Qp_cmp;
2678         _Qp_cmpe;
2679         _Qp_div;
2680         _Qp_dtoq;
2681         _Qp_feq;
2682         _Qp_fge;
2683         _Qp_fgt;
2684         _Qp_fle;
2685         _Qp_flt;
2686         _Qp_fne;
2687         _Qp_itoq;
2688         _Qp_mul;
2689         _Qp_neg;
2690         _Qp_qtod;
2691         _Qp_qtoi;
2692         _Qp_qtos;
2693         _Qp_qtoui;
2694         _Qp_qtoux;
2695         _Qp_qtox;
2696         _Qp_sqrt;
2697         _Qp_stoq;
2698         _Qp_sub;
2699         _Qp_uitoq;
2700         _Qp_uxtoq;
2701         _Qp_xtoq;
```



```
2702         __sparc_utrap_install;
2703 $endif

2705 # On amd64, we also have SYSVABI_1.3, but it contains a small subset of
2706 # the symbols put in that version on other platforms.
2707 $if amd64
2708 } SYSVABI_1.3;

2710 SYMBOL_VERSION SYSVABI_1.3 {
2711 $endif
2712     global:
2713 $if !_sparc
2714         __flt_rounds;
2715 $endif

2717     protected:
2718         _ctermid;
2719         _getgrgid;
2720         _getgrnam;
2721         _getlogin;
2722         _getpwnam;
2723         _getpwuid;
2724         _ttyname;

2726 $if !sparc32
2727         _sbrk;
2728 $endif

2730 $if _x86
2731         _fpstart;
2732         __fpstart;
2733 $endif
2734 };



2738 # There should never be more than one SUNWprivate version.
2739 # Don't add any more.  Add new private symbols to SUNWprivate_1.1

2741 SYMBOL_VERSION SUNWprivate_1.1 {
2742     global:
2743         ___Argv                 { FLAGS = NODIRECT };
2744         cfree                   { FLAGS = NODIRECT };
2745         _cswidth;
2746         __ctype_mask;
2747         __environ_lock          { FLAGS = NODIRECT };
2748         __inf_read;
2749         __inf_written;
2750         __i_size;
2751         _isnanf                 { TYPE = FUNCTION; FILTER = libm.so.2 };
2752         __iswrune;
2753         __libc_threaded;
2754         _lib_version            { FLAGS = NODIRECT };
2755         _logb                   { TYPE = FUNCTION; FILTER = libm.so.2 };
2756         _lone                   { FLAGS = NODYNSORT };
2757         _lten                   { FLAGS = NODYNSORT };
2758         _lzero                  { FLAGS = NODYNSORT };
2759         __malloc_lock;
2760         _memcmp;
2761         _memcpy                 { FLAGS = NODYNSORT };
2762         _memmove;
2763         _memset;
2764         _modff                  { TYPE = FUNCTION; FILTER = libm.so.2 };
2765         __nan_read;
2766         __nan_written;
2767         __nextwctype;
```

```
2768         __nis_debug_bind;
2769         __nis_debug_calls;
2770         __nis_debug_file;
2771         __nis_debug_rpc;
2772         __nis_prefsrv;
2773         __nis_preftype;
2774         __nis_server;
2775         _nss_default_finders;
2776         __progname              { FLAGS = NODIRECT };
2777         _smbuf;
2778         _sp;
2779         __strdupa_str           { FLAGS = NODIRECT };
2780         __strdupa_len           { FLAGS = NODIRECT };
2781         _tdb_bootstrap;
2782         __threaded;
2783         thr_probe_getfunc_addr;
2784         __trans_lower;
2785         __trans_upper;
2786         _uberdata;
2787         __xpg6                  { FLAGS = NODIRECT };

2789 $if _ELF32
2790         _dladdr                 { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2791         _dladdr1                { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2792         _dlclose                { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2793         _dldump                 { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2794         _dlerror                { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2795         _dlinfo                 { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2796         _dlmopen                { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2797         _dlopen                 { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2798         _dlsym                  { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2799         _ld_libc                { TYPE = FUNCTION; FILTER = /usr/lib/ld.so.1 };
2800         _sys_errlist;
2801         _sys_errs;
2802         _sys_index;
2803         _sys_nerr               { FLAGS = NODYNSORT };
2804         _sys_num_err;
2805 $elif sparcv9
2806         _dladdr         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2807         _dladdr1        { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2808         _dlclose        { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2809         _dldump         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2810         _dlerror        { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2811         _dlinfo         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2812         _dlmopen        { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2813         _dlopen         { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2814         _dlsym          { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2815         _ld_libc        { TYPE = FUNCTION; FILTER = /usr/lib/sparcv9/ld.so.1 };
2816 $elif amd64
2817         _dladdr         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2818         _dladdr1        { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2819         _dlamd64getunwind { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2820         _dlclose        { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2821         _dldump         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2822         _dlerror        { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2823         _dlinfo         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2824         _dlmopen        { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2825         _dlopen         { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2826         _dlsym          { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2827         _ld_libc        { TYPE = FUNCTION; FILTER = /usr/lib/amd64/ld.so.1 };
2828 $else
2829 $error unknown platform
2830 $endif

2832 $if _sparc
2833         __lyday_to_month;
```



```
2834         __mon_lengths;
2835         __yday_to_month;
2836 $endif
2837 $if i386
2838         _sse_hw;
2839 $endif

2841     protected:
2842         acctctl;
2843         allocids;
2844         _assert_c99;
2845         __assert_c99;
2846         _assfail;
2847         attr_count;
2848         attr_to_data_type;
2849         attr_to_name;
2850         attr_to_option;
2851         attr_to_xattr_view;
2852         _autofssys;
2853         _bufsync;
2854         _cladm;
2855         __class_quadruple;
2856         core_get_default_content;
2857         core_get_default_path;
2858         core_get_global_content;
2859         core_get_global_path;
2860         core_get_options;
2861         core_get_process_content;
2862         core_get_process_path;
2863         core_set_default_content;
2864         core_set_default_path;
2865         core_set_global_content;
2866         core_set_global_path;
2867         core_set_options;
2868         core_set_process_content;
2869         core_set_process_path;
2870         dbm_close_status;
2871         dbm_do_nextkey;
2872         dbm_setdefwrite;
2873         _D_cplx_div;
2874         _D_cplx_div_ix;
2875         _D_cplx_div_rx;
2876         _D_cplx_mul;
2877         defclose_r;
2878         defcntl;
2879         defcntl_r;
2880         defopen;
2881         defopen_r;
2882         defread;
2883         defread_r;
2884         _delete;
2885         _dgettext;
2886         _doprnt;
2887         _doscan;
2888         _errfp;
2889         _errxfp;
2890         exportfs;
2891         _F_cplx_div;
2892         _F_cplx_div_ix;
2893         _F_cplx_div_rx;
2894         _F_cplx_mul;
2895         __fgetwc_xpg5;
2896         __fgetws_xpg5;
2897         _findbuf;
2898         _findiop;
2899         __fini_daemon_priv;
```

```
2900          _finite;
2901          _fork1                  { FLAGS = NODYNSORT };
2902          _forkall                { FLAGS = NODYNSORT };
2903          _fpclass;
2904          _fpgetmask;
2905          _fpgetround;
2906          _fpgetsticky;
2907          _fprintf;
2908          _fpsetmask;
2909          _fpsetround;
2910          _fpsetsticky;
2911          __fputwc_xpg5;
2912          __fputws_xpg5;
2913          _ftw;
2914          _gcvt;
2915          _getarg;
2916          __getcontext;
2917          _getdents;
2918          _get_exit_frame_monitor;
2919          _getfp;
2920          _getgroupsbymember;
2921          _getlogin_r;
2922          getrandom;
2923          _getsp;
2924          __gettsp;
2925          getvmusage;
2926          __getwchar_xpg5;
2927          __getwc_xpg5;
2928          gtty;
2929          __idmap_flush_kcache;
2930          __idmap_reg;
2931          __idmap_unreg;
2932          __init_daemon_priv;
2933          __init_suid_priv;
2934          _insert;
2935          inst_sync;
2936          _iswctype;
2937          klpd_create;
2938          klpd_getpath;
2939          klpd_getport;
2940          klpd_getucred;
2941          klpd_register;
2942          klpd_register_id;
2943          klpd_unregister;
2944          klpd_unregister_id;
2945          _lgrp_home_fast         { FLAGS = NODYNSORT };
2946          _lgrpsys;
2947          _lltostr;
2948          _lock_clear;
2949          _lock_try;
2950          _ltzset;
2951          lwp_self;
2952          makeut;
2953          makeutx;
2954          _mbftowc;
2955          mcfiller;
2956          mntopt;
2957          modctl;
2958          modutx;
2959          msgctl64;
2960          __multi_innetgr;
2961          _mutex_destroy          { FLAGS = NODYNSORT };
2962          mutex_enter;
2963          mutex_exit;
2964          mutex_held;
2965          _mutex_init             { FLAGS = NODYNSORT };
```



```
2966          _mutex_unlock           { FLAGS = NODYNSORT };
2967          name_to_attr;
2968          nfs_getfh;
2969          nfssvc;
2970          _nfssys;
2971          __nis_get_environment;
2972          _nss_db_state_destr;
2973          nss_default_key2str;
2974          nss_delete;
2975          nss_endent;
2976          nss_getent;
2977          _nss_initf_group;
2978          _nss_initf_netgroup;
2979          _nss_initf_passwd;
2980          _nss_initf_shadow;
2981          nss_packed_arg_init;
2982          nss_packed_context_init;
2983          nss_packed_getkey;
2984          nss_packed_set_status;
2985          nss_search;
2986          nss_setent;
2987          _nss_XbyY_fgets;
2988          __nsw_extended_action_v1;
2989          __nsw_freeconfig_v1;
2990          __nsw_getconfig_v1;
2991          __nthreads;
2992          __openattrdirat;
2993          option_to_attr;
2994          __priv_bracket;
2995          __priv_relinquish;
2996          pset_assign_forced;
2997          pset_bind_lwp;
2998          _psignal;
2999          _pthread_setcleanupinit;
3000          __putwchar_xpg5;
3001          __putwc_xpg5;
3002          rctlctl;
3003          rctllist;
3004          _realbufend;
3005          _resume;
3006          _resume_ret;
3007          _rpcsys;
3008          _sbrk_grow_aligned;
3009          scrwidth;
3010          semctl64;
3011          _semctl64;
3012          set_setcontext_enforcement;
3013          _setbufend;
3014          __set_errno;
3015          setprojrctl;
3016          _setregid;
3017          _setreuid;
3018          setsigacthandler;
3019          shmctl64;
3020          _shmctl64;
3021          sigflag;
3022          _signal;
3023          _sigoff;
3024          _sigon;
3025          _so_accept;
3026          _so_bind;
3027          _sockconfig;
3028          _so_connect;
3029          _so_getpeername;
3030          _so_getsockname;
3031          _so_getsockopt;
```

```
3032         _so_listen;
3033         _so_recv;
3034         _so_recvfrom;
3035         _so_recvmsg;
3036         _so_send;
3037         _so_sendmsg;
3038         _so_sendto;
3039         _so_setsockopt;
3040         _so_shutdown;
3041         _so_socket;
3042         _so_socketpair;
3043         str2group;
3044         str2passwd;
3045         str2spwd;
3046         __strptime_dontzero;
3047         stty;
3048         syscall;
3049         _sysconfig;
3050         __systemcall;
3051         thr_continue_allmutators;
3052         _thr_continue_allmutators;
3053         thr_continue_mutator;
3054         _thr_continue_mutator;
3055         thr_getstate;
3056         _thr_getstate;
3057         thr_mutators_barrier;
3058         _thr_mutators_barrier;
3059         thr_probe_setup;
3060         _thr_schedctl;
3061         thr_setmutator;
3062         _thr_setmutator;
3063         thr_setstate;
3064         _thr_setstate;
3065         thr_sighndlrinfo;
3066         _thr_sighndlrinfo;
3067         _thr_slot_offset;
3068         thr_suspend_allmutators;
3069         _thr_suspend_allmutators;
3070         thr_suspend_mutator;
3071         _thr_suspend_mutator;
3072         thr_wait_mutator;
3073         _thr_wait_mutator;
3074         __tls_get_addr;
3075         _tmem_get_base;
3076         _tmem_get_nentries;
3077         _tmem_set_cleanup;
3078         tpool_create;
3079         tpool_dispatch;
3080         tpool_destroy;
3081         tpool_wait;
3082         tpool_suspend;
3083         tpool_suspended;
3084         tpool_resume;
3085         tpool_member;
3086         _ttyname_dev;
3087         _ucred_alloc;
3088         ucred_getamask;
3089         _ucred_getamask;
3090         ucred_getasid;
3091         _ucred_getasid;
3092         ucred_getatid;
3093         _ucred_getatid;
3094         ucred_getauid;
3095         _ucred_getauid;
3096         _ulltostr;
3097         _uncached_getgrgid_r;
```



```
3098         _uncached_getgrnam_r;
3099         _uncached_getpwnam_r;
3100         _uncached_getpwuid_r;
3101         __ungetwc_xpg5;
3102         _unordered;
3103         utssys;
3104         _verrfp;
3105         _verrxfp;
3106         _vwarnfp;
3107         _vwarnxfp;
3108         _warnfp;
3109         _warnxfp;
3110         __wcsftime_xpg5;
3111         __wcstok_xpg5;
3112         wdbindf;
3113         wdchkind;
3114         wddelim;
3115         _wrtchk;
3116         _xflsbuf;
3117         _xgetwidth;
3118         zone_add_datalink;
3119         zone_boot;
3120         zone_check_datalink;
3121         zone_create;
3122         zone_destroy;
3123         zone_enter;
3124         zone_getattr;
3125         zone_get_id;
3126         zone_list;
3127         zone_list_datalink;
3128         zonept;
3129         zone_remove_datalink;
3130         zone_setattr;
3131         zone_shutdown;
3132         zone_version;

3134 $if _ELF32
3135         __divdi3;
3136         _file_set;
3137         _fprintf_c89;
3138         _fscanf_c89;
3139         _fwprintf_c89;
3140         _fwscanf_c89;
3141         _imaxabs_c89;
3142         _imaxdiv_c89;
3143         __moddi3;
3144         _printf_c89;
3145         _scanf_c89;
3146         _snprintf_c89;
3147         _sprintf_c89;
3148         _sscanf_c89;
3149         _strtoimax_c89;
3150         _strtoumax_c89;
3151         _swprintf_c89;
3152         _swscanf_c89;
3153         __udivdi3;
3154         __umoddi3;
3155         _vfprintf_c89;
3156         _vfscanf_c89;
3157         _vfwprintf_c89;
3158         _vfwscanf_c89;
3159         _vprintf_c89;
3160         _vscanf_c89;
3161         _vsnprintf_c89;
3162         _vsprintf_c89;
3163         _vsscanf_c89;
```

```
3164         _vswprintf_c89;
3165         _vswscanf_c89;
3166         _vwprintf_c89;
3167         _vwscanf_c89;
3168         _wcstoimax_c89;
3169         _wcstoumax_c89;
3170         _wprintf_c89;
3171         _wscanf_c89;
3172 $endif

3174 $if _sparc
3175         _cerror;
3176         install_utrap;
3177         _install_utrap;
3178         nop;
3179         _Q_cplx_div;
3180         _Q_cplx_div_ix;
3181         _Q_cplx_div_rx;
3182         _Q_cplx_lr_div;
3183         _Q_cplx_lr_div_ix;
3184         _Q_cplx_lr_div_rx;
3185         _Q_cplx_lr_mul;
3186         _Q_cplx_mul;
3187         _QgetRD;
3188         _xregs_clrptr;
3189 $endif

3191 $if sparc32
3192         __ashldi3;
3193         __ashrdi3;
3194         _cerror64;
3195         __cmpdi2;
3196         __floatdidf;
3197         __floatdisf;
3198         __floatundidf;
3199         __floatundisf;
3200         __lshrdi3;
3201         __muldi3;
3202         __ucmpdi2;
3203 $endif

3205 $if _x86
3206         _D_cplx_lr_div;
3207         _D_cplx_lr_div_ix;
3208         _D_cplx_lr_div_rx;
3209         _F_cplx_lr_div;
3210         _F_cplx_lr_div_ix;
3211         _F_cplx_lr_div_rx;
3212         __fltrounds;
3213         sysi86;
3214         _sysi86;
3215         _X_cplx_div;
3216         _X_cplx_div_ix;
3217         _X_cplx_div_rx;
3218         _X_cplx_lr_div;
3219         _X_cplx_lr_div_ix;
3220         _X_cplx_lr_div_rx;
3221         _X_cplx_mul;
3222         __xgetRD;
3223         __xtol;
3224         __xtoll;
3225         __xtoul;
3226         __xtoull;
3227 $endif

3229 $if i386
```



```
3230         __divrem64;
3231         ___tls_get_addr;
3232         __udivrem64;
3233 $endif

3235 # The following functions should not be exported from libc,
3236 # but /lib/libm.so.2, some older versions of the Studio
3237 # compiler/debugger components, and some ancient programs
3238 # found in /usr/dist reference them.  When we no longer
3239 # care about these old and broken binary objects, these
3240 # symbols should be deleted.
3241         _brk                                    { FLAGS = NODYNSORT };
3242         _cond_broadcast                         { FLAGS = NODYNSORT };
3243         _cond_init                              { FLAGS = NODYNSORT };
3244         _cond_signal                            { FLAGS = NODYNSORT };
3245         _cond_wait                              { FLAGS = NODYNSORT };
3246         _ecvt                                   { FLAGS = NODYNSORT };
3247         _fcvt                                   { FLAGS = NODYNSORT };
3248         _getc_unlocked                          { FLAGS = NODYNSORT };
3249         _llseek                                 { FLAGS = NODYNSORT };
3250         _pthread_attr_getdetachstate            { FLAGS = NODYNSORT };
3251         _pthread_attr_getinheritsched           { FLAGS = NODYNSORT };
3252         _pthread_attr_getschedparam             { FLAGS = NODYNSORT };
3253         _pthread_attr_getschedpolicy            { FLAGS = NODYNSORT };
3254         _pthread_attr_getscope                  { FLAGS = NODYNSORT };
3255         _pthread_attr_getstackaddr              { FLAGS = NODYNSORT };
3256         _pthread_attr_getstacksize              { FLAGS = NODYNSORT };
3257         _pthread_attr_init                      { FLAGS = NODYNSORT };
3258         _pthread_condattr_getpshared            { FLAGS = NODYNSORT };
3259         _pthread_condattr_init                  { FLAGS = NODYNSORT };
3260         _pthread_cond_init                      { FLAGS = NODYNSORT };
3261         _pthread_create                         { FLAGS = NODYNSORT };
3262         _pthread_getschedparam                  { FLAGS = NODYNSORT };
3263         _pthread_join                           { FLAGS = NODYNSORT };
3264         _pthread_key_create                     { FLAGS = NODYNSORT };
3265         _pthread_mutexattr_getprioceiling       { FLAGS = NODYNSORT };
3266         _pthread_mutexattr_getprotocol          { FLAGS = NODYNSORT };
3267         _pthread_mutexattr_getpshared           { FLAGS = NODYNSORT };
3268         _pthread_mutexattr_init                 { FLAGS = NODYNSORT };
3269         _pthread_mutex_getprioceiling           { FLAGS = NODYNSORT };
3270         _pthread_mutex_init                     { FLAGS = NODYNSORT };
3271         _pthread_sigmask                        { FLAGS = NODYNSORT };
3272         _rwlock_init                            { FLAGS = NODYNSORT };
3273         _rw_rdlock                              { FLAGS = NODYNSORT };
3274         _rw_unlock                              { FLAGS = NODYNSORT };
3275         _rw_wrlock                              { FLAGS = NODYNSORT };
3276         _sbrk_unlocked                          { FLAGS = NODYNSORT };
3277         _select                                 { FLAGS = NODYNSORT };
3278         _sema_init                              { FLAGS = NODYNSORT };
3279         _sema_post                              { FLAGS = NODYNSORT };
3280         _sema_trywait                           { FLAGS = NODYNSORT };
3281         _sema_wait                              { FLAGS = NODYNSORT };
3282         _sysfs                                  { FLAGS = NODYNSORT };
3283         _thr_create                             { FLAGS = NODYNSORT };
3284         _thr_exit                               { FLAGS = NODYNSORT };
3285         _thr_getprio                            { FLAGS = NODYNSORT };
3286         _thr_getspecific                        { FLAGS = NODYNSORT };
3287         _thr_join                               { FLAGS = NODYNSORT };
3288         _thr_keycreate                          { FLAGS = NODYNSORT };
3289         _thr_kill                               { FLAGS = NODYNSORT };
3290         _thr_main                               { FLAGS = NODYNSORT };
3291         _thr_self                               { FLAGS = NODYNSORT };
3292         _thr_setspecific                        { FLAGS = NODYNSORT };
3293         _thr_sigsetmask                         { FLAGS = NODYNSORT };
3294         _thr_stksegment                         { FLAGS = NODYNSORT };
3295         _ungetc_unlocked                        { FLAGS = NODYNSORT };
```

```
3297     local:
3298         __imax_lldiv                            { FLAGS = NODYNSORT };
3299         _ti_thr_self                            { FLAGS = NODYNSORT };
3300         *;

3302 $if lf64
3303         _seekdir64              { FLAGS = NODYNSORT };
3304         _telldir64              { FLAGS = NODYNSORT };
3305 $endif

3307 $if _sparc
3308         __cerror                { FLAGS = NODYNSORT };
3309 $endif

3311 $if sparc32
3312         __cerror64              { FLAGS = NODYNSORT };
3313 $endif

3315 $if sparcv9
3316         __cleanup               { FLAGS = NODYNSORT };
3317 $endif

3319 $if i386
3320         _syscall6               { FLAGS = NODYNSORT };
3321         __systemcall6           { FLAGS = NODYNSORT };
3322 $endif

3324 $if amd64
3325         ___tls_get_addr         { FLAGS = NODYNSORT };
3326 $endif
3327 };
```

```
**********************************************************
   18165 Mon Dec 28 20:02:12 2015
new/usr/src/lib/pam_modules/unix_cred/unix_cred.c
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
______unchanged_portion_omitted_

153 /*
154  *      unix_cred - pam_sm_setcred
155  *
156  *      Entry flags =   PAM_ESTABLISH_CRED, set up Solaris Unix cred.
157  *                      PAM_DELETE_CRED, NOP, return PAM_SUCCESS.
158  *                      PAM_REINITIALIZE_CRED, set up Solaris Unix cred,
159  *                              or merge the current context with the new
160  *                              user.
161  *                      PAM_REFRESH_CRED, set up Solaris Unix cred.
162  *                      PAM_SILENT, print no messages to user.
163  *
164  *      Returns PAM_SUCCESS, if all successful.
165  *              PAM_CRED_ERR, if unable to set credentials.
166  *              PAM_USER_UNKNOWN, if PAM_USER not set, or unable to find
167  *                      user in databases.
168  *              PAM_SYSTEM_ERR, if no valid flag, or unable to get/set
169  *                      user's audit state.
170  */

172 int
173 pam_sm_setcred(pam_handle_t *pamh, int flags, int argc, const char **argv)
174 {
175         int     i;
176         int     debug = 0;
177         uint_t  nowarn = flags & PAM_SILENT;
178         int     ret = PAM_SUCCESS;
179         char    *user;
180         char    *auser;
181         char    *rhost;
182         char    *tty;
183         au_id_t auid;
184         adt_session_data_t *ah;
185         adt_termid_t    *termid = NULL;
186         priv_set_t      *lim, *def, *tset;
187         char            messages[PAM_MAX_NUM_MSG][PAM_MAX_MSG_SIZE];
188         char            buf[PROJECT_BUFSZ];
189         struct project  proj, *pproj;
190         int             error;
191         char            *projname;
192         char            *kvs;
193         struct passwd   pwd;
194         char            pwbuf[NSS_BUFLEN_PASSWD];
195         deflim_t        deflim;

197         for (i = 0; i < argc; i++) {
198                 if (strcmp(argv[i], "debug") == 0)
199                         debug = 1;
200                 else if (strcmp(argv[i], "nowarn") == 0)
201                         nowarn |= 1;
202         }

204         if (debug)
205                 syslog(LOG_AUTH | LOG_DEBUG,
206                     "pam_unix_cred: pam_sm_setcred(flags = %x, argc= %d)",
207                     flags, argc);

209         (void) pam_get_item(pamh, PAM_USER, (void **)&user);

211         if (user == NULL || *user == '\0') {
```



```
212                 syslog(LOG_AUTH | LOG_ERR,
213                     "pam_unix_cred: USER NULL or empty!\n");
214                 return (PAM_USER_UNKNOWN);
215         }
216         (void) pam_get_item(pamh, PAM_AUSER, (void **)&auser);
217         (void) pam_get_item(pamh, PAM_RHOST, (void **)&rhost);
218         (void) pam_get_item(pamh, PAM_TTY, (void **)&tty);
219         if (debug)
220                 syslog(LOG_AUTH | LOG_DEBUG,
221                     "pam_unix_cred: user = %s, auser = %s, rhost = %s, "
222                     "tty = %s", user,
223                     (auser == NULL) ? "NULL" : (*auser == '\0') ? "ZERO" :
224                     auser,
225                     (rhost == NULL) ? "NULL" : (*rhost == '\0') ? "ZERO" :
226                     rhost,
227                     (tty == NULL) ? "NULL" : (*tty == '\0') ? "ZERO" :
228                     tty);

230         /* validate flags */
231         switch (flags & (PAM_ESTABLISH_CRED | PAM_DELETE_CRED |
232             PAM_REINITIALIZE_CRED | PAM_REFRESH_CRED)) {
233         case 0:
234                 /* set default flag */
235                 flags |= PAM_ESTABLISH_CRED;
236                 break;
237         case PAM_ESTABLISH_CRED:
238         case PAM_REINITIALIZE_CRED:
239         case PAM_REFRESH_CRED:
240                 break;
241         case PAM_DELETE_CRED:
242                 return (PAM_SUCCESS);
243         default:
244                 syslog(LOG_AUTH | LOG_ERR,
245                     "pam_unix_cred: invalid flags %x", flags);
246                 return (PAM_SYSTEM_ERR);
247         }

249         /*
250          * if auditing on and process audit state not set,
251          * setup audit context for process.
252          */
253         if (adt_start_session(&ah, NULL, ADT_USE_PROC_DATA) != 0) {
254                 syslog(LOG_AUTH | LOG_ERR,
255                     "pam_unix_cred: cannot create start audit session %m");
256                 return (PAM_SYSTEM_ERR);
257         }
258         adt_get_auid(ah, &auid);
259         if (debug) {
260                 int     auditstate;

262                 if (auditon(A_GETCOND, (caddr_t)&auditstate,
263                     sizeof (auditstate)) != 0) {
264                         auditstate = AUC_DISABLED;
265                 }
266                 syslog(LOG_AUTH | LOG_DEBUG,
267                     "pam_unix_cred: state = %d, auid = %d", auditstate,
268                     auid);
269         }
270         if (getpwnam_r(user, &pwd, pwbuf, sizeof (pwbuf)) == NULL) {
271                 syslog(LOG_AUTH | LOG_ERR,
272                     "pam_unix_cred: cannot get passwd entry for user = %s",
273                     user);
274                 ret = PAM_USER_UNKNOWN;
275                 goto adt_done;
276         }
```

```
278         if ((auid == AU_NOAUDITID) &&
279             (flags & PAM_ESTABLISH_CRED)) {
280                 struct passwd   apwd;
281                 char    apwbuf[NSS_BUFLEN_PASSWD];
282
283                 errno = 0;
284                 if ((rhost == NULL || *rhost == '\0')) {
285                         if (adt_load_ttyname(tty, &termid) != 0) {
286                                 if (errno == ENETDOWN) {
287                                         /*
288                                          * tolerate not being able to
289                                          * translate local hostname
290                                          * to a termid -- it will be
291                                          * "loopback".
292                                          */
293                                         syslog(LOG_AUTH | LOG_ERR,
294                                             "pam_unix_cred: cannot load "
295                                             "ttyname: %m, continuing.");
296                                         goto adt_setuser;
297                                 } else if (errno != 0) {
298                                         syslog(LOG_AUTH | LOG_ERR,
299                                             "pam_unix_cred: cannot load "
300                                             "ttyname: %m.");
301                                 } else {
302                                         syslog(LOG_AUTH | LOG_ERR,
303                                             "pam_unix_cred: cannot load "
304                                             "ttyname.");
305                                 }
306                                 ret = PAM_SYSTEM_ERR;
307                                 goto adt_done;
308                         }
309                 } else {
310                         if (adt_load_hostname(rhost, &termid) != 0) {
311                                 if (errno != 0) {
312                                         syslog(LOG_AUTH | LOG_ERR,
313                                             "pam_unix_cred: cannot load "
314                                             "hostname: %m.");
315                                 } else {
316                                         syslog(LOG_AUTH | LOG_ERR,
317                                             "pam_unix_cred: cannot load "
318                                             "hostname.");
319                                 }
320                                 ret = PAM_SYSTEM_ERR;
321                                 goto adt_done;
322                         }
323                 }
324 adt_setuser:
325                 if ((auser != NULL) && (*auser != '\0') &&
326                     (getpwnam_r(auser, &apwd, apwbuf,
327                     sizeof (apwbuf)) != NULL)) {
328                         /*
329                          * set up the initial audit for user coming
330                          * from another user
331                          */
332                         if (adt_set_user(ah, apwd.pw_uid, apwd.pw_gid,
333                             apwd.pw_uid, apwd.pw_gid, termid, ADT_NEW) != 0) {
334                                 syslog(LOG_AUTH | LOG_ERR,
335                                     "pam_unix_cred: cannot set auser audit "
336                                     "%m");
337                                 ret = PAM_SYSTEM_ERR;
338                                 goto adt_done;
339                         }
340                         if (adt_set_user(ah, pwd.pw_uid, pwd.pw_gid,
341                             pwd.pw_uid, pwd.pw_gid, NULL,
342                             ADT_UPDATE) != 0) {
343                                 syslog(LOG_AUTH | LOG_ERR,
```



```
344                                     "pam_unix_cred: cannot merge user audit "
345                                     "%m");
346                                 ret = PAM_SYSTEM_ERR;
347                                 goto adt_done;
348                         }
349                         if (debug) {
350                                 syslog(LOG_AUTH | LOG_DEBUG,
351                                     "pam_unix_cred: new audit set for %d:%d",
352                                     apwd.pw_uid, pwd.pw_uid);
353                         }
354                 } else {
355                         /*
356                          * No authenticated user or authenticated user is
357                          * not a local user, no remote attribution, set
358                          * up the initial audit as for direct user login
359                          */
360                         if (adt_set_user(ah, pwd.pw_uid, pwd.pw_gid,
361                             pwd.pw_uid, pwd.pw_gid, termid, ADT_NEW) != 0) {
362                                 syslog(LOG_AUTH | LOG_ERR,
363                                     "pam_unix_cred: cannot set user audit %m");
364                                 ret = PAM_SYSTEM_ERR;
365                                 goto adt_done;
366                         }
367                 }
368                 if (adt_set_proc(ah) != 0) {
369                         syslog(LOG_AUTH | LOG_ERR,
370                             "pam_unix_cred: cannot set process audit %m");
371                         ret = PAM_CRED_ERR;
372                         goto adt_done;
373                 }
374                 if (debug) {
375                         syslog(LOG_AUTH | LOG_DEBUG,
376                             "pam_unix_cred: new audit set for %d",
377                             pwd.pw_uid);
378                 }
379         } else if ((auid != AU_NOAUDITID) &&
380             (flags & PAM_REINITIALIZE_CRED)) {
381                 if (adt_set_user(ah, pwd.pw_uid, pwd.pw_gid, pwd.pw_uid,
382                     pwd.pw_gid, NULL, ADT_UPDATE) != 0) {
383                         syslog(LOG_AUTH | LOG_ERR,
384                             "pam_unix_cred: cannot set user audit %m");
385                         ret = PAM_SYSTEM_ERR;
386                         goto adt_done;
387                 }
388                 if (adt_set_proc(ah) != 0) {
389                         syslog(LOG_AUTH | LOG_ERR,
390                             "pam_unix_cred: cannot set process audit %m");
391                         ret = PAM_CRED_ERR;
392                         goto adt_done;
393                 }
394                 if (debug) {
395                         syslog(LOG_AUTH | LOG_DEBUG,
396                             "pam_unix_cred: audit merged for %d:%d",
397                             auid, pwd.pw_uid);
398                 }
399         } else if (debug) {
400                 syslog(LOG_AUTH | LOG_DEBUG,
401                     "pam_unix_cred: audit already set for %d", auid);
402         }
403 adt_done:
404         if (termid != NULL)
405                 free(termid);
406         if (adt_end_session(ah) != 0) {
407                 syslog(LOG_AUTH | LOG_ERR,
408                     "pam_unix_cred: unable to end audit session");
409         }
```

```
411         if (ret != PAM_SUCCESS)
412                 return (ret);
413
414         /* Initialize the user's project */
415         (void) pam_get_item(pamh, PAM_RESOURCE, (void **)&kvs);
416         if (kvs != NULL) {
417                 char *tmp, *lasts, *tok;
418
419                 kvs = tmp = strdup(kvs);
420                 if (kvs == NULL)
421                         return (PAM_BUF_ERR);
422
423                 while ((tok = strtok_r(tmp, ";", &lasts)) != NULL) {
424                         if (strncmp(tok, PROJECT, PROJSZ) == 0) {
425                                 projname = tok + PROJSZ;
426                                 break;
427                         }
428                         tmp = NULL;
429                 }
430         } else {
431                 projname = NULL;
432         }
433
434         if (projname == NULL || *projname == '\0') {
435                 pproj = getdefaultproj(user, &proj, (void *)&buf,
436                     PROJECT_BUFSZ);
437         } else {
438                 pproj = getprojbyname(projname, &proj, (void *)&buf,
439                     PROJECT_BUFSZ);
440         }
441         /* projname points into kvs, so this is the first opportunity to free */
442         if (kvs != NULL)
443                 free(kvs);
444         if (pproj == NULL) {
445                 syslog(LOG_AUTH | LOG_ERR,
446                     "pam_unix_cred: no default project for user %s", user);
447                 if (!nowarn) {
448                         (void) snprintf(messages[0], sizeof (messages[0]),
449                             dgettext(TEXT_DOMAIN, "No default project!"));
450                         (void) __pam_display_msg(pamh, PAM_ERROR_MSG,
451                             1, messages, NULL);
452                 }
453                 return (PAM_SYSTEM_ERR);
454         }
455         if ((error = setproject(proj.pj_name, user, TASK_NORMAL)) != 0) {
456                 kva_t *kv_array;
457
458                 switch (error) {
459                 case SETPROJ_ERR_TASK:
460                         if (errno == EAGAIN) {
461                                 syslog(LOG_AUTH | LOG_ERR,
462                                     "pam_unix_cred: project \"%s\" resource "
463                                     "control limit has been reached",
464                                     proj.pj_name);
465                                 (void) snprintf(messages[0],
466                                     sizeof (messages[0]), dgettext(
467                                     TEXT_DOMAIN,
468                                     "Resource control limit has been "
469                                     "reached"));
470                         } else {
471                                 syslog(LOG_AUTH | LOG_ERR,
472                                     "pam_unix_cred: user %s could not join "
473                                     "project \"%s\": %m", user, proj.pj_name);
474                                 (void) snprintf(messages[0],
475                                     sizeof (messages[0]), dgettext(
```



```
476                                     TEXT_DOMAIN,
477                                     "Could not join default project"));
478                         }
479                         if (!nowarn)
480                                 (void) __pam_display_msg(pamh, PAM_ERROR_MSG, 1,
481                                     messages, NULL);
482                         break;
483                 case SETPROJ_ERR_POOL:
484                         (void) snprintf(messages[0], sizeof (messages[0]),
485                             dgettext(TEXT_DOMAIN,
486                             "Could not bind to resource pool"));
487                         switch (errno) {
488                         case EACCES:
489                                 syslog(LOG_AUTH | LOG_ERR,
490                                     "pam_unix_cred: project \"%s\" could not "
491                                     "bind to resource pool: No resource pool "
492                                     "accepting default bindings exists",
493                                     proj.pj_name);
494                                 (void) snprintf(messages[1],
495                                     sizeof (messages[1]),
496                                     dgettext(TEXT_DOMAIN,
497                                     "No resource pool accepting "
498                                     "default bindings exists"));
499                                 break;
500                         case ESRCH:
501                                 syslog(LOG_AUTH | LOG_ERR,
502                                     "pam_unix_cred: project \"%s\" could not "
503                                     "bind to resource pool: The resource pool "
504                                     "is unknown", proj.pj_name);
505                                 (void) snprintf(messages[1],
506                                     sizeof (messages[1]),
507                                     dgettext(TEXT_DOMAIN,
508                                     "The specified resource pool "
509                                     "is unknown"));
510                                 break;
511                         default:
512                                 (void) snprintf(messages[1],
513                                     sizeof (messages[1]),
514                                     dgettext(TEXT_DOMAIN,
515                                     "Failure during pool binding"));
516                                 syslog(LOG_AUTH | LOG_ERR,
517                                     "pam_unix_cred: project \"%s\" could not "
518                                     "bind to resource pool: %m", proj.pj_name);
519                         }
520                         if (!nowarn)
521                                 (void) __pam_display_msg(pamh, PAM_ERROR_MSG,
522                                     2, messages, NULL);
523                         break;
524                 default:
525                         /*
526                          * Resource control assignment failed.  Unlike
527                          * newtask(1m), we treat this as an error.
528                          */
529                         if (error < 0) {
530                                 /*
531                                  * This isn't supposed to happen, but in
532                                  * case it does, this error message
533                                  * doesn't use error as an index, like
534                                  * the others might.
535                                  */
536                                 syslog(LOG_AUTH | LOG_ERR,
537                                     "pam_unix_cred: unkwown error joining "
538                                     "project \"%s\" (%d)", proj.pj_name, error);
539                                 (void) snprintf(messages[0],
540                                     sizeof (messages[0]),
541                                     dgettext(TEXT_DOMAIN,
```

```
542                                     "unkwown error joining project \"%s\""
543                                     " (%d)"), proj.pj_name, error);
544                             } else if ((kv_array = _str2kva(proj.pj_attr, KV_ASSIGN,
545                                 KV_DELIMITER)) != NULL) {
546                                     syslog(LOG_AUTH | LOG_ERR,
547                                         "pam_unix_cred: %s resource control "
548                                         "assignment failed for project \"%s\"",
549                                         kv_array->data[error - 1].key,
550                                         proj.pj_name);
551                                     (void) snprintf(messages[0],
552                                         sizeof (messages[0]),
553                                         dgettext(TEXT_DOMAIN,
554                                         "%s resource control assignment failed for "
555                                         "project \"%s\""),
556                                         kv_array->data[error - 1].key,
557                                         proj.pj_name);
558                                     _kva_free(kv_array);
559                             } else {
560                                     syslog(LOG_AUTH | LOG_ERR,
561                                         "pam_unix_cred: resource control "
562                                         "assignment failed for project \"%s\""
563                                         "attribute %d", proj.pj_name, error);
564                                     (void) snprintf(messages[0],
565                                         sizeof (messages[0]),
566                                         dgettext(TEXT_DOMAIN,
567                                         "resource control assignment failed for "
568                                         "project \"%s\" attribute %d"),
569                                         proj.pj_name, error);
570                             }
571                             if (!nowarn)
572                                     (void) __pam_display_msg(pamh, PAM_ERROR_MSG,
573                                         1, messages, NULL);
574                     }
575                     return (PAM_SYSTEM_ERR);
576             }

578             tset = def = lim = NULL;
579             deflim.def = deflim.lim = NULL;

581             (void) _enum_attrs(user, finddeflim, NULL, &deflim);

583             if (getset(deflim.lim, &lim) != 0 || getset(deflim.def, &def) != 0) {
584                     ret = PAM_SYSTEM_ERR;
585                     goto out;
586             }

588             if (def == NULL) {
589                     def = priv_allocset();
590                     if (def == NULL) {
591                             ret = PAM_SYSTEM_ERR;
592                             goto out;
593                     }
594                     priv_defaultset(def);
594                     priv_basicset(def);
595                     errno = 0;
596                     if ((pathconf("/", _PC_CHOWN_RESTRICTED) == -1) && (errno == 0))
597                             (void) priv_addset(def, PRIV_FILE_CHOWN_SELF);
598             }
599             /*
600              * Silently limit the privileges to those actually available
601              * in the current zone.
602              *
603              * XXX: i think this applies to non-zone cases too, fix the comment.
604 #endif /* ! codereview */
605              */
606             tset = priv_allocset();
```



```
607             if (tset == NULL) {
608                     ret = PAM_SYSTEM_ERR;
609                     goto out;
610             }
611             if (getppriv(PRIV_PERMITTED, tset) != 0) {
612                     ret = PAM_SYSTEM_ERR;
613                     goto out;
614             }
615             if (!priv_issubset(def, tset))
616                     priv_intersect(tset, def);
617             /*
618              * We set privilege awareness here so that I gets copied to
619              * P & E when the final setuid(uid) happens.
620              */
621             (void) setpflags(PRIV_AWARE, 1);
622             if (setppriv(PRIV_SET, PRIV_INHERITABLE, def) != 0) {
623                     syslog(LOG_AUTH | LOG_ERR,
624                         "pam_setcred: setppriv(defaultpriv) failed: %m");
625                     ret = PAM_CRED_ERR;
626             }

628             if (lim != NULL) {
629                     /*
630                      * Silently limit the privileges to the limit set available.
631                      */
632                     if (getppriv(PRIV_LIMIT, tset) != 0) {
633                             ret = PAM_SYSTEM_ERR;
634                             goto out;
635                     }
636                     if (!priv_issubset(lim, tset))
637                             priv_intersect(tset, lim);
638                     if (setppriv(PRIV_SET, PRIV_LIMIT, lim) != 0) {
639                             syslog(LOG_AUTH | LOG_ERR,
640                                 "pam_setcred: setppriv(limitpriv) failed: %m");
641                             ret = PAM_CRED_ERR;
642                             goto out;
643                     }
644                     /*
645                      * In order not to surprise certain applications, we
646                      * need to get rid of privilege awareness and thus we must
647                      * set this flag which will cause a reset on set*uid().
648                      */
649                     (void) setpflags(PRIV_AWARE_RESET, 1);
650             }
651             /*
652              * This may fail but we do not care as this will be reset later
653              * when the uids are set to their final values.
654              */
655             (void) setpflags(PRIV_AWARE, 0);
656             /*
657              * Remove PRIV_PFEXEC; stop running as if we are under a profile
658              * shell.  A user with a profile shell will set PRIV_PFEXEC.
659              */
660             (void) setpflags(PRIV_PFEXEC, 0);

662 out:
663             free(deflim.lim);
664             free(deflim.def);

666             if (lim != NULL)
667                     priv_freeset(lim);
668             if (def != NULL)
669                     priv_freeset(def);
670             if (tset != NULL)
671                     priv_freeset(tset);
```

```
673         return (ret);
674 }
```

```
**********************************************************
   66839 Mon Dec 28 20:02:18 2015
new/usr/src/man/man3c/Makefile
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
   1 #
   2 # This file and its contents are supplied under the terms of the
   3 # Common Development and Distribution License ("CDDL"), version 1.0.
   4 # You may only use this file in accordance with the terms of version
   5 # 1.0 of the CDDL.
   6 #
   7 # A full copy of the text of the CDDL should have accompanied this
   8 # source.  A copy of the CDDL is also available via the Internet
   9 # at http://www.illumos.org/license/CDDL.
  10 #

  12 #
  13 # Copyright 2011, Richard Lowe
  14 # Copyright 2013 Nexenta Systems, Inc.  All rights reserved.
  15 # Copyright 2013, OmniTI Computer Consulting, Inc. All rights reserved.
  16 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
  17 # Copyright (c) 2015, Joyent, Inc.  All rights reserved.
  18 #

  20 include         $(SRC)/Makefile.master

  22 MANSECT=        3c

  24 MANFILES=       __fbufsize.3c                                   \
  25                 _longjmp.3c                                     \
  26                 _stack_grow.3c                                  \
  27                 a64l.3c                                         \
  28                 abort.3c                                        \
  29                 abs.3c                                          \
  30                 addsev.3c                                       \
  31                 addseverity.3c                                  \
  32                 aio_cancel.3c                                   \
  33                 aio_error.3c                                    \
  34                 aio_fsync.3c                                    \
  35                 aio_read.3c                                     \
  36                 aio_return.3c                                   \
  37                 aio_suspend.3c                                  \
  38                 aio_waitn.3c                                    \
  39                 aio_write.3c                                    \
  40                 aiocancel.3c                                    \
  41                 aioread.3c                                      \
  42                 aiowait.3c                                      \
  43                 arc4random.3c                                   \
  44                 assert.3c                                       \
  45                 atexit.3c                                       \
  46                 atomic_add.3c                                   \
  47                 atomic_and.3c                                   \
  48                 atomic_bits.3c                                  \
  49                 atomic_cas.3c                                   \
  50                 atomic_dec.3c                                   \
  51                 atomic_inc.3c                                   \
  52                 atomic_ops.3c                                   \
  53                 atomic_or.3c                                    \
  54                 atomic_swap.3c                                  \
  55                 attropen.3c                                     \
  56                 basename.3c                                     \
  57                 bsd_signal.3c                                   \
  58                 bsearch.3c                                      \
  59                 bstring.3c                                      \
  60                 btowc.3c                                        \
  61                 catgets.3c                                      \
```



```
  62                 catopen.3c                                      \
  63                 cfgetispeed.3c                                  \
  64                 cfsetispeed.3c                                  \
  65                 clearenv.3c                                     \
  66                 clock.3c                                        \
  67                 clock_nanosleep.3c                              \
  68                 clock_settime.3c                                \
  69                 closedir.3c                                     \
  70                 closefrom.3c                                    \
  71                 cond_init.3c                                    \
  72                 confstr.3c                                      \
  73                 crypt.3c                                        \
  74                 crypt_genhash_impl.3c                           \
  75                 crypt_gensalt.3c                                \
  76                 crypt_gensalt_impl.3c                           \
  77                 cset.3c                                         \
  78                 ctermid.3c                                      \
  79                 ctime.3c                                        \
  80                 ctype.3c                                        \
  81                 cuserid.3c                                      \
  82                 daemon.3c                                       \
  83                 decimal_to_floating.3c                          \
  84                 difftime.3c                                     \
  85                 directio.3c                                     \
  86                 dirfd.3c                                        \
  87                 dirname.3c                                      \
  88                 div.3c                                          \
  89                 dladdr.3c                                       \
  90                 dlclose.3c                                      \
  91                 dldump.3c                                       \
  92                 dlerror.3c                                      \
  93                 dlinfo.3c                                       \
  94                 dlopen.3c                                       \
  95                 dlsym.3c                                        \
  96                 door_bind.3c                                    \
  97                 door_call.3c                                    \
  98                 door_create.3c                                  \
  99                 door_cred.3c                                    \
 100                 door_getparam.3c                                \
 101                 door_info.3c                                    \
 102                 door_return.3c                                  \
 103                 door_revoke.3c                                  \
 104                 door_server_create.3c                           \
 105                 door_ucred.3c                                   \
 106                 drand48.3c                                      \
 107                 dup2.3c                                         \
 108                 econvert.3c                                     \
 109                 ecvt.3c                                         \
 110                 enable_extended_FILE_stdio.3c                   \
 111                 encrypt.3c                                      \
 112                 end.3c                                          \
 113                 epoll_create.3c                                 \
 114                 epoll_ctl.3c                                    \
 115                 epoll_wait.3c                                   \
 116                 err.3c                                          \
 117                 euclen.3c                                       \
 118                 eventfd.3c                                      \
 119                 exit.3c                                         \
 120                 fattach.3c                                      \
 121                 fclose.3c                                       \
 122                 fcloseall.3c                                    \
 123                 fdatasync.3c                                    \
 124                 fdetach.3c                                      \
 125                 fdopen.3c                                       \
 126                 ferror.3c                                       \
 127                 fflush.3c                                       \
```

```
128                     ffs.3c                                  \
129                     fgetattr.3c                             \
130                     fgetc.3c                                \
131                     fgetpos.3c                              \
132                     fgetwc.3c                               \
133                     floating_to_decimal.3c                  \
134                     flock.3c                                \
135                     flockfile.3c                            \
136                     fmtmsg.3c                               \
137                     fnmatch.3c                              \
138                     fopen.3c                                \
139                     fpgetround.3c                           \
140                     fputc.3c                                \
141                     fputwc.3c                               \
142                     fputws.3c                               \
143                     fread.3c                                \
144                     freopen.3c                              \
145                     fseek.3c                                \
146                     fsetpos.3c                              \
147                     fsync.3c                                \
148                     ftell.3c                                \
149                     ftime.3c                                \
150                     ftok.3c                                 \
151                     ftw.3c                                  \
152                     fwide.3c                                \
153                     fwprintf.3c                             \
154                     fwrite.3c                               \
155                     fwscanf.3c                              \
156                     getcpuid.3c                             \
157                     getcwd.3c                               \
158                     getdate.3c                              \
159                     getdtablesize.3c                        \
160                     getentropy.3c                           \
161                     getenv.3c                               \
162                     getexecname.3c                          \
163                     getgrnam.3c                             \
164                     gethostid.3c                            \
165                     gethostname.3c                          \
166                     gethrtime.3c                            \
167                     getline.3c                              \
168                     getloadavg.3c                           \
169                     getlogin.3c                             \
170                     getmntent.3c                            \
171                     getnetgrent.3c                          \
172                     get_nprocs.3c                           \
173                     getopt.3c                               \
174                     getpagesize.3c                          \
175                     getpagesizes.3c                         \
176                     getpass.3c                              \
177                     getpeerucred.3c                         \
178                     getpriority.3c                          \
179                     getprogname.3c                          \
180                     getpw.3c                                \
181                     getpwnam.3c                             \
182                     getrusage.3c                            \
183                     gets.3c                                 \
184                     getspnam.3c                             \
185                     getsubopt.3c                            \
186                     gettext.3c                              \
187                     gettimeofday.3c                         \
188                     gettxt.3c                               \
189                     getusershell.3c                         \
190                     getutent.3c                             \
191                     getutxent.3c                            \
192                     getvfsent.3c                            \
193                     getwc.3c                                \
```



```
194                     getwchar.3c                             \
195                     getwd.3c                                \
196                     getwidth.3c                             \
197                     getws.3c                                \
198                     getzoneid.3c                            \
199                     glob.3c                                 \
200                     grantpt.3c                              \
201                     hsearch.3c                              \
202                     iconv.3c                                \
203                     iconv_close.3c                          \
204                     iconv_open.3c                           \
205                     imaxabs.3c                              \
206                     imaxdiv.3c                              \
207                     index.3c                                \
208                     initgroups.3c                           \
209                     insque.3c                               \
210                     is_system_labeled.3c                    \
211                     isaexec.3c                              \
212                     isastream.3c                            \
213                     isatty.3c                               \
214                     isnand.3c                               \
215                     iswalpha.3c                             \
216                     iswctype.3c                             \
217                     killpg.3c                               \
218                     lckpwdf.3c                              \
219                     lfmt.3c                                 \
220                     lio_listio.3c                           \
221                     localeconv.3c                           \
222                     lockf.3c                                \
223                     lsearch.3c                              \
224                     madvise.3c                              \
225                     makecontext.3c                          \
226                     makedev.3c                              \
227                     malloc.3c                               \
228                     mblen.3c                                \
229                     mbrlen.3c                               \
230                     mbrtowc.3c                              \
231                     mbsinit.3c                              \
232                     mbsrtowcs.3c                            \
233                     mbtowc.3c                               \
234                     membar_ops.3c                           \
235                     memory.3c                               \
236                     mkfifo.3c                               \
237                     mkstemp.3c                              \
238                     mktemp.3c                               \
239                     mktime.3c                               \
240                     mlock.3c                                \
241                     mlockall.3c                             \
242                     monitor.3c                              \
243                     mq_close.3c                             \
244                     mq_getattr.3c                           \
245                     mq_notify.3c                            \
246                     mq_open.3c                              \
247                     mq_receive.3c                           \
248                     mq_send.3c                              \
249                     mq_setattr.3c                           \
250                     mq_unlink.3c                            \
251                     msync.3c                                \
252                     mutex_init.3c                           \
253                     nanosleep.3c                            \
254                     ndbm.3c                                 \
255                     newlocale.3c                            \
256                     nl_langinfo.3c                          \
257                     offsetof.3c                             \
258                     opendir.3c                              \
259                     perror.3c                               \
```

```
260                pfmt.3c                                      \
261                plock.3c                                     \
262                popen.3c                                     \
263                port_alert.3c                                \
264                port_associate.3c                            \
265                port_create.3c                               \
266                port_get.3c                                  \
267                port_send.3c                                 \
268                posix_fadvise.3c                             \
269                posix_fallocate.3c                           \
270                posix_madvise.3c                             \
271                posix_memalign.3c                            \
272                posix_openpt.3c                              \
273                posix_spawn.3c                               \
274                posix_spawn_file_actions_addclose.3c         \
275                posix_spawn_file_actions_addclosefrom_np.3c  \
276                posix_spawn_file_actions_adddup2.3c          \
277                posix_spawn_file_actions_destroy.3c          \
278                posix_spawn_pipe_np.3c                       \
279                posix_spawnattr_destroy.3c                   \
280                posix_spawnattr_getflags.3c                  \
281                posix_spawnattr_getpgroup.3c                 \
282                posix_spawnattr_getschedparam.3c             \
283                posix_spawnattr_getschedpolicy.3c            \
284                posix_spawnattr_getsigdefault.3c             \
285                posix_spawnattr_getsigignore_np.3c           \
286                posix_spawnattr_getsigmask.3c                \
287                printf.3c                                    \
288                priv_addset.3c                               \
289                priv_set.3c                                  \
290                priv_str_to_set.3c                           \
291                pset_getloadavg.3c                           \
292                psignal.3c                                   \
293                pthread_atfork.3c                            \
294                pthread_attr_getdetachstate.3c               \
295                pthread_attr_getguardsize.3c                 \
296                pthread_attr_getinheritsched.3c              \
297                pthread_attr_getschedparam.3c                \
298                pthread_attr_getschedpolicy.3c               \
299                pthread_attr_getscope.3c                     \
300                pthread_attr_getstack.3c                     \
301                pthread_attr_getstackaddr.3c                 \
302                pthread_attr_getstacksize.3c                 \
303                pthread_attr_init.3c                         \
304                pthread_barrier_destroy.3c                   \
305                pthread_barrier_wait.3c                      \
306                pthread_barrierattr_destroy.3c               \
307                pthread_barrierattr_getpshared.3c            \
308                pthread_cancel.3c                            \
309                pthread_cleanup_pop.3c                       \
310                pthread_cleanup_push.3c                      \
311                pthread_cond_init.3c                         \
312                pthread_cond_signal.3c                       \
313                pthread_cond_wait.3c                         \
314                pthread_condattr_getclock.3c                 \
315                pthread_condattr_getpshared.3c               \
316                pthread_condattr_init.3c                     \
317                pthread_create.3c                            \
318                pthread_detach.3c                            \
319                pthread_equal.3c                             \
320                pthread_exit.3c                              \
321                pthread_getconcurrency.3c                    \
322                pthread_getschedparam.3c                     \
323                pthread_getspecific.3c                       \
324                pthread_join.3c                              \
325                pthread_key_create.3c                        \
```



```
326                pthread_key_delete.3c                        \
327                pthread_kill.3c                              \
328                pthread_mutex_consistent.3c                  \
329                pthread_mutex_getprioceiling.3c              \
330                pthread_mutex_init.3c                        \
331                pthread_mutex_lock.3c                        \
332                pthread_mutex_timedlock.3c                   \
333                pthread_mutexattr_getprioceiling.3c          \
334                pthread_mutexattr_getprotocol.3c             \
335                pthread_mutexattr_getpshared.3c              \
336                pthread_mutexattr_getrobust.3c               \
337                pthread_mutexattr_gettype.3c                 \
338                pthread_mutexattr_init.3c                    \
339                pthread_once.3c                              \
340                pthread_rwlock_init.3c                       \
341                pthread_rwlock_rdlock.3c                     \
342                pthread_rwlock_timedrdlock.3c                \
343                pthread_rwlock_timedwrlock.3c                \
344                pthread_rwlock_unlock.3c                     \
345                pthread_rwlock_wrlock.3c                     \
346                pthread_rwlockattr_getpshared.3c             \
347                pthread_rwlockattr_init.3c                   \
348                pthread_self.3c                              \
349                pthread_setcancelstate.3c                    \
350                pthread_setcanceltype.3c                     \
351                pthread_setschedprio.3c                      \
352                pthread_sigmask.3c                           \
353                pthread_spin_destroy.3c                      \
354                pthread_spin_lock.3c                         \
355                pthread_spin_unlock.3c                       \
356                pthread_testcancel.3c                        \
357                ptrace.3c                                    \
358                ptsname.3c                                   \
359                putenv.3c                                    \
360                putpwent.3c                                  \
361                puts.3c                                      \
362                putspent.3c                                  \
363                putws.3c                                     \
364                qsort.3c                                     \
365                raise.3c                                     \
366                rand.3c                                      \
367                random.3c                                    \
368                rctl_walk.3c                                 \
369                rctlblk_set_value.3c                         \
370                re_comp.3c                                   \
371                readdir.3c                                   \
372                realpath.3c                                  \
373                reboot.3c                                    \
374                regcmp.3c                                    \
375                regcomp.3c                                   \
376                remove.3c                                    \
377                rewind.3c                                    \
378                rewinddir.3c                                 \
379                rwlock.3c                                    \
380                scandir.3c                                   \
381                scanf.3c                                     \
382                sched_get_priority_max.3c                    \
383                sched_getparam.3c                            \
384                sched_getscheduler.3c                        \
385                sched_rr_get_interval.3c                     \
386                sched_setparam.3c                            \
387                sched_setscheduler.3c                        \
388                sched_yield.3c                               \
389                schedctl_init.3c                             \
390                seekdir.3c                                   \
391                select.3c                                    \
```

```
392                     sem_close.3c                                    \
393                     sem_destroy.3c                                  \
394                     sem_getvalue.3c                                 \
395                     sem_init.3c                                     \
396                     sem_open.3c                                     \
397                     sem_post.3c                                     \
398                     sem_timedwait.3c                                \
399                     sem_unlink.3c                                   \
400                     sem_wait.3c                                     \
401                     semaphore.3c                                    \
402                     setbuf.3c                                       \
403                     setbuffer.3c                                    \
404                     setcat.3c                                       \
405                     setenv.3c                                       \
406                     setjmp.3c                                       \
407                     setkey.3c                                       \
408                     setlabel.3c                                     \
409                     setlocale.3c                                    \
410                     shm_open.3c                                     \
411                     shm_unlink.3c                                   \
412                     sigfpe.3c                                       \
413                     siginterrupt.3c                                 \
414                     signal.3c                                       \
415                     signalfd.3c                                     \
416                     sigqueue.3c                                     \
417                     sigsetops.3c                                    \
418                     sigstack.3c                                     \
419                     sigwaitinfo.3c                                  \
420                     sleep.3c                                        \
421                     ssignal.3c                                      \
422                     smt_pause.3c                                    \
423                     stack_getbounds.3c                              \
424                     stack_inbounds.3c                               \
425                     stack_setbounds.3c                              \
426                     stack_violation.3c                              \
427                     stdio.3c                                        \
428                     str2sig.3c                                      \
429                     strcoll.3c                                      \
430                     strerror.3c                                     \
431                     strfmon.3c                                      \
432                     strftime.3c                                     \
433                     string.3c                                       \
434                     string_to_decimal.3c                            \
435                     strptime.3c                                     \
436                     strsignal.3c                                    \
437                     strtod.3c                                       \
438                     strtoimax.3c                                    \
439                     strtol.3c                                       \
440                     strtoul.3c                                      \
441                     strtows.3c                                      \
442                     strxfrm.3c                                      \
443                     swab.3c                                         \
444                     sync_instruction_memory.3c                      \
445                     sysconf.3c                                      \
446                     syslog.3c                                       \
447                     system.3c                                       \
448                     tcdrain.3c                                      \
449                     tcflow.3c                                       \
450                     tcflush.3c                                      \
451                     tcgetattr.3c                                    \
452                     tcgetpgrp.3c                                    \
453                     tcgetsid.3c                                     \
454                     tcsendbreak.3c                                  \
455                     tcsetattr.3c                                    \
456                     tcsetpgrp.3c                                    \
457                     tell.3c                                         \
```



```
458                     telldir.3c                                      \
459                     termios.3c                                      \
460                     thr_create.3c                                   \
461                     thr_exit.3c                                     \
462                     thr_getconcurrency.3c                           \
463                     thr_getprio.3c                                  \
464                     thr_join.3c                                     \
465                     thr_keycreate.3c                                \
466                     thr_kill.3c                                     \
467                     thr_main.3c                                     \
468                     thr_min_stack.3c                                \
469                     thr_self.3c                                     \
470                     thr_sigsetmask.3c                               \
471                     thr_stksegment.3c                               \
472                     thr_suspend.3c                                  \
473                     thr_yield.3c                                    \
474                     timer_create.3c                                 \
475                     timer_delete.3c                                 \
476                     timer_settime.3c                                \
477                     timeradd.3c                                     \
478                     timerfd_create.3c                               \
479                     tmpfile.3c                                      \
480                     tmpnam.3c                                       \
481                     toascii.3c                                      \
482                     tolower.3c                                      \
483                     toupper.3c                                      \
484                     towlower.3c                                     \
485                     towupper.3c                                     \
486                     truncate.3c                                     \
487                     tsearch.3c                                      \
488                     ttyname.3c                                      \
489                     ttyslot.3c                                      \
490                     u8_strcmp.3c                                    \
491                     u8_textprep_str.3c                              \
492                     u8_validate.3c                                  \
493                     ualarm.3c                                       \
494                     uconv_u16tou32.3c                               \
495                     ucred.3c                                        \
496                     ungetc.3c                                       \
497                     ungetwc.3c                                      \
498                     unlockpt.3c                                     \
499                     unsetenv.3c                                     \
500                     uselocale.3c                                    \
501                     usleep.3c                                       \
502                     vfwprintf.3c                                    \
503                     vlfmt.3c                                        \
504                     vpfmt.3c                                        \
505                     vprintf.3c                                      \
506                     vsyslog.3c                                      \
507                     wait.3c                                         \
508                     wait3.3c                                        \
509                     waitpid.3c                                      \
510                     walkcontext.3c                                  \
511                     wcpcpy.3c                                       \
512                     wcrtomb.3c                                      \
513                     wcscasecmp.3c                                   \
514                     wcscoll.3c                                      \
515                     wcsdup.3c                                       \
516                     wcslen.3c                                       \
517                     wcsftime.3c                                     \
518                     wcsrtombs.3c                                    \
519                     wcsstr.3c                                       \
520                     wcstod.3c                                       \
521                     wcstoimax.3c                                    \
522                     wcstol.3c                                       \
523                     wcstoul.3c                                      \
```

```
524                 wcstring.3c                                 \
525                 wcswidth.3c                                 \
526                 wcsxfrm.3c                                  \
527                 wctob.3c                                    \
528                 wctomb.3c                                   \
529                 wctrans.3c                                  \
530                 wctype.3c                                   \
531                 wcwidth.3c                                  \
532                 wmemchr.3c                                  \
533                 wmemcmp.3c                                  \
534                 wmemcpy.3c                                  \
535                 wmemmove.3c                                 \
536                 wmemset.3c                                  \
537                 wordexp.3c                                  \
538                 wsprintf.3c                                 \
539                 wsscanf.3c                                  \
540                 wstring.3c                                  \

542 MANLINKS=       FD_CLR.3c                           \
543                 FD_ISSET.3c                         \
544                 FD_SET.3c                           \
545                 FD_ZERO.3c                          \
546                 __flbf.3c                           \
547                 __fpending.3c                       \
548                 __fpurge.3c                         \
549                 __freadable.3c                      \
550                 __freading.3c                       \
551                 __fsetlocking.3c                    \
552                 __fwritable.3c                      \
553                 __fwriting.3c                       \
554                 _edata.3c                           \
555                 _end.3c                             \
556                 _etext.3c                           \
557                 _exithandle.3c                      \
558                 _flushlbf.3c                        \
559                 _setjmp.3c                          \
560                 addrtosymstr.3c                     \
561                 aiowrite.3c                         \
562                 alloca.3c                           \
563                 alphasort.3c                        \
564                 ascftime.3c                         \
565                 asctime.3c                          \
566                 asctime_r.3c                        \
567                 asprintf.3c                         \
568                 atof.3c                             \
569                 atoi.3c                             \
570                 atol.3c                             \
571                 atoll.3c                            \
572                 arc4random_buf.3c                   \
573                 arc4random_uniform.3c               \
574                 atomic_add_16.3c                    \
575                 atomic_add_16_nv.3c                 \
576                 atomic_add_32.3c                    \
577                 atomic_add_32_nv.3c                 \
578                 atomic_add_64.3c                    \
579                 atomic_add_64_nv.3c                 \
580                 atomic_add_8.3c                     \
581                 atomic_add_8_nv.3c                  \
582                 atomic_add_char.3c                  \
583                 atomic_add_char_nv.3c               \
584                 atomic_add_int.3c                   \
585                 atomic_add_int_nv.3c                \
586                 atomic_add_long.3c                  \
587                 atomic_add_long_nv.3c               \
588                 atomic_add_ptr.3c                   \
589                 atomic_add_ptr_nv.3c                \
```



```
590                 atomic_add_short.3c                 \
591                 atomic_add_short_nv.3c              \
592                 atomic_and_16.3c                    \
593                 atomic_and_16_nv.3c                 \
594                 atomic_and_32.3c                    \
595                 atomic_and_32_nv.3c                 \
596                 atomic_and_64.3c                    \
597                 atomic_and_64_nv.3c                 \
598                 atomic_and_8.3c                     \
599                 atomic_and_8_nv.3c                  \
600                 atomic_and_uchar.3c                 \
601                 atomic_and_uchar_nv.3c              \
602                 atomic_and_uint.3c                  \
603                 atomic_and_uint_nv.3c               \
604                 atomic_and_ulong.3c                 \
605                 atomic_and_ulong_nv.3c              \
606                 atomic_and_ushort.3c                \
607                 atomic_and_ushort_nv.3c             \
608                 atomic_cas_16.3c                    \
609                 atomic_cas_32.3c                    \
610                 atomic_cas_64.3c                    \
611                 atomic_cas_8.3c                     \
612                 atomic_cas_ptr.3c                   \
613                 atomic_cas_uchar.3c                 \
614                 atomic_cas_uint.3c                  \
615                 atomic_cas_ulong.3c                 \
616                 atomic_cas_ushort.3c                \
617                 atomic_clear_long_excl.3c           \
618                 atomic_dec_16.3c                    \
619                 atomic_dec_16_nv.3c                 \
620                 atomic_dec_32.3c                    \
621                 atomic_dec_32_nv.3c                 \
622                 atomic_dec_64.3c                    \
623                 atomic_dec_64_nv.3c                 \
624                 atomic_dec_8.3c                     \
625                 atomic_dec_8_nv.3c                  \
626                 atomic_dec_ptr.3c                   \
627                 atomic_dec_ptr_nv.3c                \
628                 atomic_dec_uchar.3c                 \
629                 atomic_dec_uchar_nv.3c              \
630                 atomic_dec_uint.3c                  \
631                 atomic_dec_uint_nv.3c               \
632                 atomic_dec_ulong.3c                 \
633                 atomic_dec_ulong_nv.3c              \
634                 atomic_dec_ushort.3c                \
635                 atomic_dec_ushort_nv.3c             \
636                 atomic_inc_16.3c                    \
637                 atomic_inc_16_nv.3c                 \
638                 atomic_inc_32.3c                    \
639                 atomic_inc_32_nv.3c                 \
640                 atomic_inc_64.3c                    \
641                 atomic_inc_64_nv.3c                 \
642                 atomic_inc_8.3c                     \
643                 atomic_inc_8_nv.3c                  \
644                 atomic_inc_ptr.3c                   \
645                 atomic_inc_ptr_nv.3c                \
646                 atomic_inc_uchar.3c                 \
647                 atomic_inc_uchar_nv.3c              \
648                 atomic_inc_uint.3c                  \
649                 atomic_inc_uint_nv.3c               \
650                 atomic_inc_ulong.3c                 \
651                 atomic_inc_ulong_nv.3c              \
652                 atomic_inc_ushort.3c                \
653                 atomic_inc_ushort_nv.3c             \
654                 atomic_or_16.3c                     \
655                 atomic_or_16_nv.3c                  \
```

```
656                 atomic_or_32.3c                 \
657                 atomic_or_32_nv.3c              \
658                 atomic_or_64.3c                 \
659                 atomic_or_64_nv.3c              \
660                 atomic_or_8.3c                  \
661                 atomic_or_8_nv.3c               \
662                 atomic_or_uchar.3c              \
663                 atomic_or_uchar_nv.3c           \
664                 atomic_or_uint.3c               \
665                 atomic_or_uint_nv.3c            \
666                 atomic_or_ulong.3c              \
667                 atomic_or_ulong_nv.3c           \
668                 atomic_or_ushort.3c             \
669                 atomic_or_ushort_nv.3c          \
670                 atomic_set_long_excl.3c         \
671                 atomic_swap_16.3c               \
672                 atomic_swap_32.3c               \
673                 atomic_swap_64.3c               \
674                 atomic_swap_8.3c                \
675                 atomic_swap_ptr.3c              \
676                 atomic_swap_uchar.3c            \
677                 atomic_swap_uint.3c             \
678                 atomic_swap_ulong.3c            \
679                 atomic_swap_ushort.3c           \
680                 backtrace.3c                    \
681                 backtrace_symbols.3c            \
682                 backtrace_symbols_fd.3c         \
683                 bcmp.3c                         \
684                 bcopy.3c                        \
685                 bind_textdomain_codeset.3c      \
686                 bindtextdomain.3c               \
687                 btowc_l.3c                      \
688                 bzero.3c                        \
689                 calloc.3c                       \
690                 canonicalize_file_name.3c       \
691                 catclose.3c                     \
692                 cfgetospeed.3c                  \
693                 cfsetospeed.3c                  \
694                 cftime.3c                       \
695                 clearerr.3c                     \
696                 clock_getres.3c                 \
697                 clock_gettime.3c                \
698                 closelog.3c                     \
699                 cond_broadcast.3c               \
700                 cond_destroy.3c                 \
701                 cond_reltimedwait.3c            \
702                 cond_signal.3c                  \
703                 cond_timedwait.3c               \
704                 cond_wait.3c                    \
705                 csetcol.3c                      \
706                 csetlen.3c                      \
707                 csetno.3c                       \
708                 ctermid_r.3c                    \
709                 ctime_r.3c                      \
710                 dbm_clearerr.3c                 \
711                 dbm_close.3c                    \
712                 dbm_delete.3c                   \
713                 dbm_error.3c                    \
714                 dbm_fetch.3c                    \
715                 dbm_firstkey.3c                 \
716                 dbm_nextkey.3c                  \
717                 dbm_open.3c                     \
718                 dbm_store.3c                    \
719                 dcgettext.3c                    \
720                 dcngettext.3c                   \
721                 decimal_to_double.3c            \
```



```
722                 decimal_to_extended.3c          \
723                 decimal_to_quadruple.3c         \
724                 decimal_to_single.3c            \
725                 dgettext.3c                     \
726                 dladdr1.3c                      \
727                 dlmopen.3c                      \
728                 dngettext.3c                    \
729                 door_setparam.3c                \
730                 door_unbind.3c                  \
731                 double_to_decimal.3c            \
732                 dup3.3c                         \
733                 duplocale.3c                    \
734                 edata.3c                        \
735                 endgrent.3c                     \
736                 endnetgrent.3c                  \
737                 endpwent.3c                     \
738                 endspent.3c                     \
739                 endusershell.3c                 \
740                 endutent.3c                     \
741                 endutxent.3c                    \
742                 epoll_create1.3c                \
743                 epoll_pwait.3c                  \
744                 erand48.3c                      \
745                 errno.3c                        \
746                 errx.3c                         \
747                 etext.3c                        \
748                 euccol.3c                       \
749                 eucscol.3c                      \
750                 explicit_bzero.3c               \
751                 extended_to_decimal.3c          \
752                 fconvert.3c                     \
753                 fcvt.3c                         \
754                 fdopendir.3c                    \
755                 fdwalk.3c                       \
756                 feof.3c                         \
757                 ffsl.3c                         \
758                 ffsll.3c                        \
759                 fgetgrent.3c                    \
760                 fgetgrent_r.3c                  \
761                 fgetpwent.3c                    \
762                 fgetpwent_r.3c                  \
763                 fgets.3c                        \
764                 fgetspent.3c                    \
765                 fgetspent_r.3c                  \
766                 fgetwc_l.3c                     \
767                 fgetws.3c                       \
768                 file_to_decimal.3c              \
769                 fileno.3c                       \
770                 finite.3c                       \
771                 fls.3c                          \
772                 flsl.3c                         \
773                 flsll.3c                        \
774                 fpclass.3c                      \
775                 fpgetmask.3c                    \
776                 fpgetsticky.3c                  \
777                 fprintf.3c                      \
778                 fpsetmask.3c                    \
779                 fpsetround.3c                   \
780                 fpsetsticky.3c                  \
781                 fputs.3c                        \
782                 free.3c                         \
783                 freelocale.3c                   \
784                 fscanf.3c                       \
785                 fseeko.3c                       \
786                 fsetattr.3c                     \
787                 ftello.3c                       \
```

```
788                ftruncate.3c                        \
789                ftrylockfile.3c                     \
790                func_to_decimal.3c                  \
791                funlockfile.3c                      \
792                gconvert.3c                         \
793                gcvt.3c                             \
794                getattrat.3c                        \
795                getc.3c                             \
796                getc_unlocked.3c                    \
797                getchar.3c                          \
798                getchar_unlocked.3c                 \
799                getdelim.3c                         \
800                getextmntent.3c                     \
801                getgrent.3c                         \
802                getgrent_r.3c                       \
803                getgrgid.3c                         \
804                getgrgid_r.3c                       \
805                getgrnam_r.3c                       \
806                gethomelgroup.3c                    \
807                gethrvtime.3c                       \
808                getlogin_r.3c                       \
809                getmntany.3c                        \
810                getnetgrent_r.3c                    \
811                get_nprocs_conf.3c                  \
812                getpassphrase.3c                    \
813                getpwent.3c                         \
814                getpwent_r.3c                       \
815                getpwnam_r.3c                       \
816                getpwuid.3c                         \
817                getpwuid_r.3c                       \
818                getspent.3c                         \
819                getspent_r.3c                       \
820                getspnam_r.3c                       \
821                getutid.3c                          \
822                getutline.3c                        \
823                getutmp.3c                          \
824                getutmpx.3c                         \
825                getutxid.3c                         \
826                getutxline.3c                       \
827                getvfsany.3c                        \
828                getvfsfile.3c                       \
829                getvfsspec.3c                       \
830                getw.3c                             \
831                getwc_l.3c                          \
832                getwchar_l.3c                       \
833                getzoneidbyname.3c                  \
834                getzonenamebyid.3c                  \
835                globfree.3c                         \
836                gmtime.3c                           \
837                gmtime_r.3c                         \
838                gsignal.3c                          \
839                hasmntopt.3c                        \
840                hcreate.3c                          \
841                hdestroy.3c                         \
842                initstate.3c                        \
843                innetgr.3c                          \
844                isalnum.3c                          \
845                isalnum_l.3c                        \
846                isalpha.3c                          \
847                isalpha_l.3c                        \
848                isascii.3c                          \
849                isblank.3c                          \
850                isblank_l.3c                        \
851                iscntrl.3c                          \
852                iscntrl_l.3c                        \
853                isdigit.3c                          \
```



```
854                isdigit_l.3c                        \
855                isenglish.3c                        \
856                isgraph.3c                          \
857                isgraph_l.3c                        \
858                isideogram.3c                       \
859                islower.3c                          \
860                islower_l.3c                        \
861                isnanf.3c                           \
862                isnumber.3c                         \
863                isphonogram.3c                      \
864                isprint.3c                          \
865                isprint_l.3c                        \
866                ispunct.3c                          \
867                ispunct_l.3c                        \
868                isspace.3c                          \
869                isspace_l.3c                        \
870                isspecial.3c                        \
871                isupper.3c                          \
872                isupper_l.3c                        \
873                iswalnum.3c                         \
874                iswalnum_l.3c                       \
875                iswalpha_l.3c                       \
876                iswascii.3c                         \
877                iswblank.3c                         \
878                iswblank_l.3c                       \
879                iswcntrl.3c                         \
880                iswcntrl_l.3c                       \
881                iswctype_l.3c                       \
882                iswdigit.3c                         \
883                iswdigit_l.3c                       \
884                iswgraph.3c                         \
885                iswgraph_l.3c                       \
886                iswideogram.3c                      \
887                iswideogram_l.3c                    \
888                iswhexnumber.3c                     \
889                iswhexnumber_l.3c                   \
890                iswlower.3c                         \
891                iswlower_l.3c                       \
892                iswnumber.3c                        \
893                iswnumber_l.3c                      \
894                iswphonogram.3c                     \
895                iswphonogram_l.3c                   \
896                iswprint.3c                         \
897                iswprint_l.3c                       \
898                iswpunct.3c                         \
899                iswpunct_l.3c                       \
900                iswspace.3c                         \
901                iswspace_l.3c                       \
902                iswspecial.3c                       \
903                iswspecial_l.3c                     \
904                iswupper.3c                         \
905                iswupper_l.3c                       \
906                iswxdigit.3c                        \
907                iswxdigit_l.3c                      \
908                isxdigit.3c                         \
909                isxdigit_l.3c                       \
910                jrand48.3c                          \
911                l64a.3c                             \
912                labs.3c                             \
913                lcong48.3c                          \
914                ldiv.3c                             \
915                lfind.3c                            \
916                llabs.3c                            \
917                lldiv.3c                            \
918                lltostr.3c                          \
919                localtime.3c                        \
```

```
920                 localtime_r.3c                          \
921                 longjmp.3c                              \
922                 lrand48.3c                              \
923                 major.3c                                \
924                 mblen_l.3c                              \
925                 mbrlen_l.3c                             \
926                 mbrtowc_l.3c                            \
927                 mbsinit_l.3c                            \
928                 mbsnrtowcs.3c                           \
929                 mbsnrtowcs_l.3c                         \
930                 mbsrtowcs_l.3c                          \
931                 mbstowcs.3c                             \
932                 mbstowcs_l.3c                           \
933                 mbtowc_l.3c                             \
934                 memalign.3c                             \
935                 membar_consumer.3c                      \
936                 membar_enter.3c                         \
937                 membar_exit.3c                          \
938                 membar_producer.3c                      \
939                 memccpy.3c                              \
940                 memchr.3c                               \
941                 memcmp.3c                               \
942                 memcpy.3c                               \
943                 memmem.3c                               \
944                 memmove.3c                              \
945                 memset.3c                               \
946                 minor.3c                                \
947                 mkdtemp.3c                              \
948                 mkfifoat.3c                             \
949                 mkostemp.3c                             \
950                 mkostemps.3c                            \
951                 mkstemps.3c                             \
952                 mq_reltimedreceive_np.3c                \
953                 mq_reltimedsend_np.3c                   \
954                 mq_timedreceive.3c                      \
955                 mq_timedsend.3c                         \
956                 mrand48.3c                              \
957                 munlock.3c                              \
958                 munlockall.3c                           \
959                 mutex_consistent.3c                     \
960                 mutex_destroy.3c                        \
961                 mutex_lock.3c                           \
962                 mutex_trylock.3c                        \
963                 mutex_unlock.3c                         \
964                 nl_langinfo_l.3c                        \
965                 nftw.3c                                 \
966                 ngettext.3c                             \
967                 nrand48.3c                              \
968                 openlog.3c                              \
969                 pclose.3c                               \
970                 port_dissociate.3c                      \
971                 port_getn.3c                            \
972                 port_sendn.3c                           \
973                 posix_spawn_file_actions_addopen.3c     \
974                 posix_spawn_file_actions_init.3c        \
975                 posix_spawnattr_init.3c                 \
976                 posix_spawnattr_setflags.3c             \
977                 posix_spawnattr_setpgroup.3c            \
978                 posix_spawnattr_setschedparam.3c        \
979                 posix_spawnattr_setschedpolicy.3c       \
980                 posix_spawnattr_setsigdefault.3c        \
981                 posix_spawnattr_setsigignore_np.3c      \
982                 posix_spawnattr_setsigmask.3c           \
983                 posix_spawnp.3c                         \
984                 printstack.3c                           \
985                 priv_allocset.3c                        \
```



```
986                 priv_basicset.3c                        \
987                 priv_copyset.3c                         \
988                 priv_defaultset.3c                      \
989 #endif /* ! codereview */
990                 priv_delset.3c                          \
991                 priv_emptyset.3c                        \
992                 priv_fillset.3c                         \
993                 priv_freeset.3c                         \
994                 priv_getbyname.3c                       \
995                 priv_getbynum.3c                        \
996                 priv_getsetbyname.3c                    \
997                 priv_getsetbynum.3c                     \
998                 priv_gettext.3c                         \
999                 priv_ineffect.3c                        \
1000                priv_intersect.3c                       \
1001                priv_inverse.3c                         \
1002                priv_isemptyset.3c                      \
1003                priv_isequalset.3c                      \
1004                priv_isfullset.3c                       \
1005                priv_ismember.3c                        \
1006                priv_issubset.3c                        \
1007                priv_set_to_str.3c                      \
1008                priv_union.3c                           \
1009                pselect.3c                              \
1010                psiginfo.3c                             \
1011                pthread_attr_destroy.3c                 \
1012                pthread_attr_setdetachstate.3c          \
1013                pthread_attr_setguardsize.3c            \
1014                pthread_attr_setinheritsched.3c         \
1015                pthread_attr_setschedparam.3c           \
1016                pthread_attr_setschedpolicy.3c          \
1017                pthread_attr_setscope.3c                \
1018                pthread_attr_setstack.3c                \
1019                pthread_attr_setstackaddr.3c            \
1020                pthread_attr_setstacksize.3c            \
1021                pthread_barrier_init.3c                 \
1022                pthread_barrierattr_init.3c             \
1023                pthread_barrierattr_setpshared.3c       \
1024                pthread_cond_broadcast.3c               \
1025                pthread_cond_destroy.3c                 \
1026                pthread_cond_reltimedwait_np.3c         \
1027                pthread_cond_timedwait.3c               \
1028                pthread_condattr_destroy.3c             \
1029                pthread_condattr_setclock.3c            \
1030                pthread_condattr_setpshared.3c          \
1031                pthread_key_create_once_np.3c           \
1032                pthread_mutex_destroy.3c                \
1033                pthread_mutex_reltimedlock_np.3c        \
1034                pthread_mutex_setprioceiling.3c         \
1035                pthread_mutex_trylock.3c                \
1036                pthread_mutex_unlock.3c                 \
1037                pthread_mutexattr_destroy.3c            \
1038                pthread_mutexattr_setprioceiling.3c     \
1039                pthread_mutexattr_setprotocol.3c        \
1040                pthread_mutexattr_setpshared.3c         \
1041                pthread_mutexattr_setrobust.3c          \
1042                pthread_mutexattr_settype.3c            \
1043                pthread_rwlock_destroy.3c               \
1044                pthread_rwlock_reltimedrdlock_np.3c     \
1045                pthread_rwlock_reltimedwrlock_np.3c     \
1046                pthread_rwlock_tryrdlock.3c             \
1047                pthread_rwlock_trywrlock.3c             \
1048                pthread_rwlockattr_destroy.3c           \
1049                pthread_rwlockattr_setpshared.3c        \
1050                pthread_setconcurrency.3c               \
1051                pthread_setschedparam.3c                \
```

```
1052                pthread_setspecific.3c              \
1053                pthread_spin_init.3c                \
1054                pthread_spin_trylock.3c             \
1055                putc.3c                             \
1056                putc_unlocked.3c                    \
1057                putchar.3c                          \
1058                putchar_unlocked.3c                 \
1059                putmntent.3c                        \
1060                pututline.3c                        \
1061                pututxline.3c                       \
1062                putw.3c                             \
1063                putwc.3c                            \
1064                putwchar.3c                         \
1065                qeconvert.3c                        \
1066                qfconvert.3c                        \
1067                qgconvert.3c                        \
1068                quadruple_to_decimal.3c             \
1069                rand_r.3c                           \
1070                rctlblk_get_enforced_value.3c       \
1071                rctlblk_get_firing_time.3c          \
1072                rctlblk_get_global_action.3c        \
1073                rctlblk_get_global_flags.3c         \
1074                rctlblk_get_local_action.3c         \
1075                rctlblk_get_local_flags.3c          \
1076                rctlblk_get_privilege.3c            \
1077                rctlblk_get_recipient_pid.3c        \
1078                rctlblk_get_value.3c                \
1079                rctlblk_set_local_action.3c         \
1080                rctlblk_set_local_flags.3c          \
1081                rctlblk_set_privilege.3c            \
1082                rctlblk_set_recipient_pid.3c        \
1083                rctlblk_size.3c                     \
1084                re_exec.3c                          \
1085                readdir_r.3c                        \
1086                realloc.3c                          \
1087                regerror.3c                         \
1088                regex.3c                            \
1089                regexec.3c                          \
1090                regfree.3c                          \
1091                remque.3c                           \
1092                resetmnttab.3c                      \
1093                rindex.3c                           \
1094                rw_rdlock.3c                        \
1095                rw_tryrdlock.3c                     \
1096                rw_trywrlock.3c                     \
1097                rw_unlock.3c                        \
1098                rw_wrlock.3c                        \
1099                rwlock_destroy.3c                   \
1100                rwlock_init.3c                      \
1101                sched_get_priority_min.3c           \
1102                schedctl_exit.3c                    \
1103                schedctl_lookup.3c                  \
1104                schedctl_start.3c                   \
1105                schedctl_stop.3c                    \
1106                seconvert.3c                        \
1107                seed48.3c                           \
1108                sem_reltimedwait_np.3c              \
1109                sem_trywait.3c                      \
1110                sema_destroy.3c                     \
1111                sema_init.3c                        \
1112                sema_post.3c                        \
1113                sema_trywait.3c                     \
1114                sema_wait.3c                        \
1115                setattrat.3c                        \
1116                setgrent.3c                         \
1117                sethostname.3c                      \
```



```
1118                setlinebuf.3c                       \
1119                setlogmask.3c                       \
1120                setnetgrent.3c                      \
1121                setpriority.3c                      \
1122                setprogname.3c                      \
1123                setpwent.3c                         \
1124                setspent.3c                         \
1125                setstate.3c                         \
1126                settimeofday.3c                     \
1127                setusershell.3c                     \
1128                setutent.3c                         \
1129                setutxent.3c                        \
1130                setvbuf.3c                          \
1131                sfconvert.3c                        \
1132                sgconvert.3c                        \
1133                sig2str.3c                          \
1134                sigaddset.3c                        \
1135                sigdelset.3c                        \
1136                sigemptyset.3c                      \
1137                sigfillset.3c                       \
1138                sighold.3c                          \
1139                sigignore.3c                        \
1140                sigismember.3c                      \
1141                siglongjmp.3c                       \
1142                sigpause.3c                         \
1143                sigrelse.3c                         \
1144                sigset.3c                           \
1145                sigsetjmp.3c                        \
1146                sigtimedwait.3c                     \
1147                single_to_decimal.3c                \
1148                snprintf.3c                         \
1149                sprintf.3c                          \
1150                srand.3c                            \
1151                srand48.3c                          \
1152                srandom.3c                          \
1153                sscanf.3c                           \
1154                stderr.3c                           \
1155                stdin.3c                            \
1156                stdout.3c                           \
1157                stpcpy.3c                           \
1158                stpncpy.3c                          \
1159                strcasecmp.3c                       \
1160                strcasecmp_l.3c                     \
1161                strcasestr.3c                       \
1162                strcasestr_l.3c                     \
1163                strcat.3c                           \
1164                strchr.3c                           \
1165                strchrnul.3c                        \
1166                strcmp.3c                           \
1167                strcoll_l.3c                        \
1168                strcpy.3c                           \
1169                strcspn.3c                          \
1170                strdup.3c                           \
1171                strdupa.3c                          \
1172                strerror_l.3c                       \
1173                strerror_r.3c                       \
1174                strfmon_l.3c                        \
1175                strftime_l.3c                       \
1176                strlcat.3c                          \
1177                strlcpy.3c                          \
1178                strlen.3c                           \
1179                strncasecmp.3c                      \
1180                strncasecmp_l.3c                    \
1181                strncat.3c                          \
1182                strncmp.3c                          \
1183                strncpy.3c                          \
```

```
1184                    strndup.3c                              \
1185                    strndupa.3c                             \
1186                    strnlen.3c                              \
1187                    strnstr.3c                              \
1188                    strpbrk.3c                              \
1189                    strptime_l.3c                           \
1190                    strrchr.3c                              \
1191                    strsep.3c                               \
1192                    strspn.3c                               \
1193                    strstr.3c                               \
1194                    strtof.3c                               \
1195                    strtok.3c                               \
1196                    strtok_r.3c                             \
1197                    strtold.3c                              \
1198                    strtoll.3c                              \
1199                    strtoull.3c                             \
1200                    strtoumax.3c                            \
1201                    strxfrm_l.3c                            \
1202                    swapcontext.3c                          \
1203                    swprintf.3c                             \
1204                    swscanf.3c                              \
1205                    tdelete.3c                              \
1206                    tempnam.3c                              \
1207                    textdomain.3c                           \
1208                    tfind.3c                                \
1209                    thr_continue.3c                         \
1210                    thr_getspecific.3c                      \
1211                    thr_keycreate_once.3c                   \
1212                    thr_setconcurrency.3c                   \
1213                    thr_setprio.3c                          \
1214                    thr_setspecific.3c                      \
1215                    timer_getoverrun.3c                     \
1216                    timer_gettime.3c                        \
1217                    timerclear.3c                           \
1218                    timercmp.3c                             \
1219                    timerfd_gettime.3c                      \
1220                    timerfd_settime.3c                      \
1221                    timerisset.3c                           \
1222                    timersub.3c                             \
1223                    tmpnam_r.3c                             \
1224                    tolower_l.3c                            \
1225                    toupper_l.3c                            \
1226                    towctrans.3c                            \
1227                    towctrans_l.3c                          \
1228                    towlower_l.3c                           \
1229                    towupper_l.3c                           \
1230                    ttyname_r.3c                            \
1231                    twalk.3c                                \
1232                    tzset.3c                                \
1233                    uconv_u16tou8.3c                        \
1234                    uconv_u32tou16.3c                       \
1235                    uconv_u32tou8.3c                        \
1236                    uconv_u8tou16.3c                        \
1237                    uconv_u8tou32.3c                        \
1238                    ucred_free.3c                           \
1239                    ucred_get.3c                            \
1240                    ucred_getegid.3c                        \
1241                    ucred_geteuid.3c                        \
1242                    ucred_getgroups.3c                      \
1243                    ucred_getlabel.3c                       \
1244                    ucred_getpflags.3c                      \
1245                    ucred_getpid.3c                         \
1246                    ucred_getprivset.3c                     \
1247                    ucred_getprojid.3c                      \
1248                    ucred_getrgid.3c                        \
1249                    ucred_getruid.3c                        \
```



```
1250                    ucred_getsgid.3c                        \
1251                    ucred_getsuid.3c                        \
1252                    ucred_getzoneid.3c                      \
1253                    ucred_size.3c                           \
1254                    ulckpwdf.3c                             \
1255                    ulltostr.3c                             \
1256                    unordered.3c                            \
1257                    updwtmp.3c                              \
1258                    updwtmpx.3c                             \
1259                    utmpname.3c                             \
1260                    utmpxname.3c                            \
1261                    valloc.3c                               \
1262                    vasprintf.3c                            \
1263                    verr.3c                                 \
1264                    verrx.3c                                \
1265                    vfprintf.3c                             \
1266                    vfscanf.3c                              \
1267                    vfwscanf.3c                             \
1268                    vscanf.3c                               \
1269                    vsnprintf.3c                            \
1270                    vsprintf.3c                             \
1271                    vsscanf.3c                              \
1272                    vswprintf.3c                            \
1273                    vswscanf.3c                             \
1274                    vwarn.3c                                \
1275                    vwarnx.3c                               \
1276                    vwprintf.3c                             \
1277                    vwscanf.3c                              \
1278                    wait4.3c                                \
1279                    warn.3c                                 \
1280                    warnx.3c                                \
1281                    watof.3c                                \
1282                    watoi.3c                                \
1283                    watol.3c                                \
1284                    watoll.3c                               \
1285                    wcpncpy.3c                              \
1286                    wcrtomb_l.3c                            \
1287                    wcscasecmp_l.3c                         \
1288                    wcscat.3c                               \
1289                    wcschr.3c                               \
1290                    wcscmp.3c                               \
1291                    wcscoll_l.3c                            \
1292                    wcscpy.3c                               \
1293                    wcscspn.3c                              \
1294                    wcsetno.3c                              \
1295                    wcsncasecmp.3c                          \
1296                    wcsncasecmp_l.3c                        \
1297                    wcsncat.3c                              \
1298                    wcsncmp.3c                              \
1299                    wcsncpy.3c                              \
1300                    wcsnlen.3c                              \
1301                    wcsnrtombs.3c                           \
1302                    wcsnrtombs_l.3c                         \
1303                    wcspbrk.3c                              \
1304                    wcsrchr.3c                              \
1305                    wcsrtombs_l.3c                          \
1306                    wcsspn.3c                               \
1307                    wcstof.3c                               \
1308                    wcstok.3c                               \
1309                    wcstold.3c                              \
1310                    wcstoll.3c                              \
1311                    wcstoull.3c                             \
1312                    wcstoumax.3c                            \
1313                    wcswcs.3c                               \
1314                    wcswidth_l.3c                           \
1315                    wctob_l.3c                              \
```

```
1316                wctomb_l.3c                             \
1317                wctrans_l.3c                            \
1318                wctype_l.3c                             \
1319                wcwidth_l.3c                            \
1320                windex.3c                               \
1321                wordfree.3c                             \
1322                wprintf.3c                              \
1323                wrindex.3c                              \
1324                wscanf.3c                               \
1325                wscasecmp.3c                            \
1326                wscat.3c                                \
1327                wschr.3c                                \
1328                wscmp.3c                                \
1329                wscol.3c                                \
1330                wscoll.3c                               \
1331                wscpy.3c                                \
1332                wscspn.3c                               \
1333                wsdup.3c                                \
1334                wslen.3c                                \
1335                wsncasecmp.3c                           \
1336                wsncat.3c                               \
1337                wsncmp.3c                               \
1338                wsncpy.3c                               \
1339                wspbrk.3c                               \
1340                wsrchr.3c                               \
1341                wsspn.3c                                \
1342                wstod.3c                                \
1343                wstok.3c                                \
1344                wstol.3c                                \
1345                wstostr.3c                              \
1346                wsxfrm.3c

1348 __flbf.3c                          := LINKSRC = __fbufsize.3c
1349 __fpending.3c                      := LINKSRC = __fbufsize.3c
1350 __fpurge.3c                        := LINKSRC = __fbufsize.3c
1351 __freadable.3c                     := LINKSRC = __fbufsize.3c
1352 __freading.3c                      := LINKSRC = __fbufsize.3c
1353 __fsetlocking.3c                   := LINKSRC = __fbufsize.3c
1354 __fwritable.3c                     := LINKSRC = __fbufsize.3c
1355 __fwriting.3c                      := LINKSRC = __fbufsize.3c
1356 _flushlbf.3c                       := LINKSRC = __fbufsize.3c

1358 _setjmp.3c                         := LINKSRC = _longjmp.3c

1360 l64a.3c                            := LINKSRC = a64l.3c

1362 labs.3c                            := LINKSRC = abs.3c
1363 llabs.3c                           := LINKSRC = abs.3c

1365 aiowrite.3c                        := LINKSRC = aioread.3c

1367 arc4random_buf.3c                  := LINKSRC = arc4random.3c
1368 arc4random_uniform.3c              := LINKSRC = arc4random.3c

1370 atomic_add_16.3c                   := LINKSRC = atomic_add.3c
1371 atomic_add_16_nv.3c                := LINKSRC = atomic_add.3c
1372 atomic_add_32.3c                   := LINKSRC = atomic_add.3c
1373 atomic_add_32_nv.3c                := LINKSRC = atomic_add.3c
1374 atomic_add_64.3c                   := LINKSRC = atomic_add.3c
1375 atomic_add_64_nv.3c                := LINKSRC = atomic_add.3c
1376 atomic_add_8.3c                    := LINKSRC = atomic_add.3c
1377 atomic_add_8_nv.3c                 := LINKSRC = atomic_add.3c
1378 atomic_add_char.3c                 := LINKSRC = atomic_add.3c
1379 atomic_add_char_nv.3c              := LINKSRC = atomic_add.3c
1380 atomic_add_int.3c                  := LINKSRC = atomic_add.3c
1381 atomic_add_int_nv.3c               := LINKSRC = atomic_add.3c
```



```
1382 atomic_add_long.3c                := LINKSRC = atomic_add.3c
1383 atomic_add_long_nv.3c             := LINKSRC = atomic_add.3c
1384 atomic_add_ptr.3c                 := LINKSRC = atomic_add.3c
1385 atomic_add_ptr_nv.3c              := LINKSRC = atomic_add.3c
1386 atomic_add_short.3c               := LINKSRC = atomic_add.3c
1387 atomic_add_short_nv.3c            := LINKSRC = atomic_add.3c
1388 atomic_and_16.3c                  := LINKSRC = atomic_and.3c
1389 atomic_and_16_nv.3c               := LINKSRC = atomic_and.3c
1390 atomic_and_32.3c                  := LINKSRC = atomic_and.3c
1391 atomic_and_32_nv.3c               := LINKSRC = atomic_and.3c
1392 atomic_and_64.3c                  := LINKSRC = atomic_and.3c
1393 atomic_and_64_nv.3c               := LINKSRC = atomic_and.3c
1394 atomic_and_8.3c                   := LINKSRC = atomic_and.3c
1395 atomic_and_8_nv.3c                := LINKSRC = atomic_and.3c
1396 atomic_and_uchar.3c               := LINKSRC = atomic_and.3c
1397 atomic_and_uchar_nv.3c            := LINKSRC = atomic_and.3c
1398 atomic_and_uint.3c                := LINKSRC = atomic_and.3c
1399 atomic_and_uint_nv.3c             := LINKSRC = atomic_and.3c
1400 atomic_and_ulong.3c               := LINKSRC = atomic_and.3c
1401 atomic_and_ulong_nv.3c            := LINKSRC = atomic_and.3c
1402 atomic_and_ushort.3c              := LINKSRC = atomic_and.3c
1403 atomic_and_ushort_nv.3c           := LINKSRC = atomic_and.3c

1405 atomic_clear_long_excl.3c         := LINKSRC = atomic_bits.3c
1406 atomic_set_long_excl.3c           := LINKSRC = atomic_bits.3c

1408 atomic_cas_16.3c                  := LINKSRC = atomic_cas.3c
1409 atomic_cas_32.3c                  := LINKSRC = atomic_cas.3c
1410 atomic_cas_64.3c                  := LINKSRC = atomic_cas.3c
1411 atomic_cas_8.3c                   := LINKSRC = atomic_cas.3c
1412 atomic_cas_ptr.3c                 := LINKSRC = atomic_cas.3c
1413 atomic_cas_uchar.3c               := LINKSRC = atomic_cas.3c
1414 atomic_cas_uint.3c                := LINKSRC = atomic_cas.3c
1415 atomic_cas_ulong.3c               := LINKSRC = atomic_cas.3c
1416 atomic_cas_ushort.3c              := LINKSRC = atomic_cas.3c

1418 atomic_dec_16.3c                  := LINKSRC = atomic_dec.3c
1419 atomic_dec_16_nv.3c               := LINKSRC = atomic_dec.3c
1420 atomic_dec_32.3c                  := LINKSRC = atomic_dec.3c
1421 atomic_dec_32_nv.3c               := LINKSRC = atomic_dec.3c
1422 atomic_dec_64.3c                  := LINKSRC = atomic_dec.3c
1423 atomic_dec_64_nv.3c               := LINKSRC = atomic_dec.3c
1424 atomic_dec_8.3c                   := LINKSRC = atomic_dec.3c
1425 atomic_dec_8_nv.3c                := LINKSRC = atomic_dec.3c
1426 atomic_dec_ptr.3c                 := LINKSRC = atomic_dec.3c
1427 atomic_dec_ptr_nv.3c              := LINKSRC = atomic_dec.3c
1428 atomic_dec_uchar.3c               := LINKSRC = atomic_dec.3c
1429 atomic_dec_uchar_nv.3c            := LINKSRC = atomic_dec.3c
1430 atomic_dec_uint.3c                := LINKSRC = atomic_dec.3c
1431 atomic_dec_uint_nv.3c             := LINKSRC = atomic_dec.3c
1432 atomic_dec_ulong.3c               := LINKSRC = atomic_dec.3c
1433 atomic_dec_ulong_nv.3c            := LINKSRC = atomic_dec.3c
1434 atomic_dec_ushort.3c              := LINKSRC = atomic_dec.3c
1435 atomic_dec_ushort_nv.3c           := LINKSRC = atomic_dec.3c

1437 atomic_inc_16.3c                  := LINKSRC = atomic_inc.3c
1438 atomic_inc_16_nv.3c               := LINKSRC = atomic_inc.3c
1439 atomic_inc_32.3c                  := LINKSRC = atomic_inc.3c
1440 atomic_inc_32_nv.3c               := LINKSRC = atomic_inc.3c
1441 atomic_inc_64.3c                  := LINKSRC = atomic_inc.3c
1442 atomic_inc_64_nv.3c               := LINKSRC = atomic_inc.3c
1443 atomic_inc_8.3c                   := LINKSRC = atomic_inc.3c
1444 atomic_inc_8_nv.3c                := LINKSRC = atomic_inc.3c
1445 atomic_inc_ptr.3c                 := LINKSRC = atomic_inc.3c
1446 atomic_inc_ptr_nv.3c              := LINKSRC = atomic_inc.3c
1447 atomic_inc_uchar.3c               := LINKSRC = atomic_inc.3c
```

```
1448 atomic_inc_uchar_nv.3c                  := LINKSRC = atomic_inc.3c
1449 atomic_inc_uint.3c                      := LINKSRC = atomic_inc.3c
1450 atomic_inc_uint_nv.3c                   := LINKSRC = atomic_inc.3c
1451 atomic_inc_ulong.3c                     := LINKSRC = atomic_inc.3c
1452 atomic_inc_ulong_nv.3c                  := LINKSRC = atomic_inc.3c
1453 atomic_inc_ushort.3c                    := LINKSRC = atomic_inc.3c
1454 atomic_inc_ushort_nv.3c                 := LINKSRC = atomic_inc.3c

1456 atomic_or_16.3c                         := LINKSRC = atomic_or.3c
1457 atomic_or_16_nv.3c                      := LINKSRC = atomic_or.3c
1458 atomic_or_32.3c                         := LINKSRC = atomic_or.3c
1459 atomic_or_32_nv.3c                      := LINKSRC = atomic_or.3c
1460 atomic_or_64.3c                         := LINKSRC = atomic_or.3c
1461 atomic_or_64_nv.3c                      := LINKSRC = atomic_or.3c
1462 atomic_or_8.3c                          := LINKSRC = atomic_or.3c
1463 atomic_or_8_nv.3c                       := LINKSRC = atomic_or.3c
1464 atomic_or_uchar.3c                      := LINKSRC = atomic_or.3c
1465 atomic_or_uchar_nv.3c                   := LINKSRC = atomic_or.3c
1466 atomic_or_uint.3c                       := LINKSRC = atomic_or.3c
1467 atomic_or_uint_nv.3c                    := LINKSRC = atomic_or.3c
1468 atomic_or_ulong.3c                      := LINKSRC = atomic_or.3c
1469 atomic_or_ulong_nv.3c                   := LINKSRC = atomic_or.3c
1470 atomic_or_ushort.3c                     := LINKSRC = atomic_or.3c
1471 atomic_or_ushort_nv.3c                  := LINKSRC = atomic_or.3c

1473 atomic_swap_16.3c                       := LINKSRC = atomic_swap.3c
1474 atomic_swap_32.3c                       := LINKSRC = atomic_swap.3c
1475 atomic_swap_64.3c                       := LINKSRC = atomic_swap.3c
1476 atomic_swap_8.3c                        := LINKSRC = atomic_swap.3c
1477 atomic_swap_ptr.3c                      := LINKSRC = atomic_swap.3c
1478 atomic_swap_uchar.3c                    := LINKSRC = atomic_swap.3c
1479 atomic_swap_uint.3c                     := LINKSRC = atomic_swap.3c
1480 atomic_swap_ulong.3c                    := LINKSRC = atomic_swap.3c
1481 atomic_swap_ushort.3c                   := LINKSRC = atomic_swap.3c

1483 bcmp.3c                                 := LINKSRC = bstring.3c
1484 bcopy.3c                                := LINKSRC = bstring.3c
1485 bzero.3c                                := LINKSRC = bstring.3c
1486 explicit_bzero.3c                       := LINKSRC = bstring.3c

1488 btowc_l.3c                              := LINKSRC = btowc.3c

1490 canonicalize_file_name.3c               := LINKSRC = realpath.3c

1492 catclose.3c                             := LINKSRC = catopen.3c

1494 cfgetospeed.3c                          := LINKSRC = cfgetispeed.3c

1496 cfsetospeed.3c                          := LINKSRC = cfsetispeed.3c

1498 clock_getres.3c                        := LINKSRC = clock_settime.3c
1499 clock_gettime.3c                       := LINKSRC = clock_settime.3c

1501 fdwalk.3c                              := LINKSRC = closefrom.3c

1503 cond_broadcast.3c                      := LINKSRC = cond_init.3c
1504 cond_destroy.3c                        := LINKSRC = cond_init.3c
1505 cond_reltimedwait.3c                   := LINKSRC = cond_init.3c
1506 cond_signal.3c                         := LINKSRC = cond_init.3c
1507 cond_timedwait.3c                      := LINKSRC = cond_init.3c
1508 cond_wait.3c                           := LINKSRC = cond_init.3c

1510 csetcol.3c                             := LINKSRC = cset.3c
1511 csetlen.3c                             := LINKSRC = cset.3c
1512 csetno.3c                              := LINKSRC = cset.3c
1513 wcsetno.3c                             := LINKSRC = cset.3c
```



```
1515 ctermid_r.3c                           := LINKSRC = ctermid.3c

1517 asctime.3c                            := LINKSRC = ctime.3c
1518 asctime_r.3c                          := LINKSRC = ctime.3c
1519 ctime_r.3c                           := LINKSRC = ctime.3c
1520 gmtime.3c                            := LINKSRC = ctime.3c
1521 gmtime_r.3c                          := LINKSRC = ctime.3c
1522 localtime.3c                         := LINKSRC = ctime.3c
1523 localtime_r.3c                       := LINKSRC = ctime.3c
1524 tzset.3c                            := LINKSRC = ctime.3c

1526 isalnum.3c                           := LINKSRC = ctype.3c
1527 isalnum_l.3c                         := LINKSRC = ctype.3c
1528 isalpha.3c                           := LINKSRC = ctype.3c
1529 isalpha_l.3c                         := LINKSRC = ctype.3c
1530 isascii.3c                           := LINKSRC = ctype.3c
1531 isblank.3c                           := LINKSRC = ctype.3c
1532 isblank_l.3c                         := LINKSRC = ctype.3c
1533 iscntrl.3c                           := LINKSRC = ctype.3c
1534 iscntrl_l.3c                         := LINKSRC = ctype.3c
1535 isdigit.3c                           := LINKSRC = ctype.3c
1536 isdigit_l.3c                         := LINKSRC = ctype.3c
1537 isgraph.3c                           := LINKSRC = ctype.3c
1538 isgraph_l.3c                         := LINKSRC = ctype.3c
1539 islower.3c                           := LINKSRC = ctype.3c
1540 islower_l.3c                         := LINKSRC = ctype.3c
1541 isprint.3c                           := LINKSRC = ctype.3c
1542 isprint_l.3c                         := LINKSRC = ctype.3c
1543 ispunct.3c                           := LINKSRC = ctype.3c
1544 ispunct_l.3c                         := LINKSRC = ctype.3c
1545 isspace.3c                           := LINKSRC = ctype.3c
1546 isspace_l.3c                         := LINKSRC = ctype.3c
1547 isupper.3c                           := LINKSRC = ctype.3c
1548 isupper_l.3c                         := LINKSRC = ctype.3c
1549 isxdigit.3c                          := LINKSRC = ctype.3c
1550 isxdigit_l.3c                        := LINKSRC = ctype.3c

1552 decimal_to_double.3c                 := LINKSRC = decimal_to_floating.3c
1553 decimal_to_extended.3c               := LINKSRC = decimal_to_floating.3c
1554 decimal_to_quadruple.3c              := LINKSRC = decimal_to_floating.3c
1555 decimal_to_single.3c                 := LINKSRC = decimal_to_floating.3c

1557 ldiv.3c                              := LINKSRC = div.3c
1558 lldiv.3c                             := LINKSRC = div.3c

1560 dladdr1.3c                           := LINKSRC = dladdr.3c

1562 dlmopen.3c                           := LINKSRC = dlopen.3c

1564 door_unbind.3c                       := LINKSRC = door_bind.3c

1566 door_setparam.3c                     := LINKSRC = door_getparam.3c

1568 erand48.3c                           := LINKSRC = drand48.3c
1569 jrand48.3c                           := LINKSRC = drand48.3c
1570 lcong48.3c                           := LINKSRC = drand48.3c
1571 lrand48.3c                           := LINKSRC = drand48.3c
1572 mrand48.3c                           := LINKSRC = drand48.3c
1573 nrand48.3c                           := LINKSRC = drand48.3c
1574 seed48.3c                            := LINKSRC = drand48.3c
1575 srand48.3c                           := LINKSRC = drand48.3c

1577 dup3.3c                              := LINKSRC = dup2.3c

1579 fconvert.3c                          := LINKSRC = econvert.3c
```

```
1580 gconvert.3c                        := LINKSRC = econvert.3c
1581 qeconvert.3c                       := LINKSRC = econvert.3c
1582 qfconvert.3c                       := LINKSRC = econvert.3c
1583 qgconvert.3c                       := LINKSRC = econvert.3c
1584 seconvert.3c                       := LINKSRC = econvert.3c
1585 sfconvert.3c                       := LINKSRC = econvert.3c
1586 sgconvert.3c                       := LINKSRC = econvert.3c

1588 fcvt.3c                            := LINKSRC = ecvt.3c
1589 gcvt.3c                            := LINKSRC = ecvt.3c

1591 _edata.3c                          := LINKSRC = end.3c
1592 _end.3c                            := LINKSRC = end.3c
1593 _etext.3c                          := LINKSRC = end.3c
1594 edata.3c                           := LINKSRC = end.3c
1595 etext.3c                           := LINKSRC = end.3c

1597 epoll_create1.3c                   := LINKSRC = epoll_create.3c
1598 epoll_pwait.3c                     := LINKSRC = epoll_wait.3c

1600 errx.3c                           := LINKSRC = err.3c
1601 verr.3c                           := LINKSRC = err.3c
1602 verrx.3c                          := LINKSRC = err.3c
1603 vwarn.3c                          := LINKSRC = err.3c
1604 vwarnx.3c                         := LINKSRC = err.3c
1605 warn.3c                           := LINKSRC = err.3c
1606 warnx.3c                          := LINKSRC = err.3c

1608 euccol.3c                         := LINKSRC = euclen.3c
1609 eucscol.3c                        := LINKSRC = euclen.3c

1611 _exithandle.3c                    := LINKSRC = exit.3c

1613 clearerr.3c                       := LINKSRC = ferror.3c
1614 feof.3c                           := LINKSRC = ferror.3c
1615 fileno.3c                         := LINKSRC = ferror.3c

1617 ffsl.3c                           := LINKSRC = ffs.3c
1618 ffsll.3c                          := LINKSRC = ffs.3c
1619 fls.3c                            := LINKSRC = ffs.3c
1620 flsl.3c                           := LINKSRC = ffs.3c
1621 flsll.3c                          := LINKSRC = ffs.3c

1623 fsetattr.3c                       := LINKSRC = fgetattr.3c
1624 getattrat.3c                      := LINKSRC = fgetattr.3c
1625 setattrat.3c                      := LINKSRC = fgetattr.3c

1627 getc.3c                           := LINKSRC = fgetc.3c
1628 getc_unlocked.3c                  := LINKSRC = fgetc.3c
1629 getchar.3c                        := LINKSRC = fgetc.3c
1630 getchar_unlocked.3c               := LINKSRC = fgetc.3c
1631 getw.3c                           := LINKSRC = fgetc.3c

1633 fgetwc_l.3c                       := LINKSRC = fgetwc.3c

1635 double_to_decimal.3c              := LINKSRC = floating_to_decimal.3c
1636 extended_to_decimal.3c            := LINKSRC = floating_to_decimal.3c
1637 quadruple_to_decimal.3c           := LINKSRC = floating_to_decimal.3c
1638 single_to_decimal.3c              := LINKSRC = floating_to_decimal.3c

1640 ftrylockfile.3c                   := LINKSRC = flockfile.3c
1641 funlockfile.3c                    := LINKSRC = flockfile.3c

1643 fpgetmask.3c                      := LINKSRC = fpgetround.3c
1644 fpgetsticky.3c                    := LINKSRC = fpgetround.3c
1645 fpsetmask.3c                      := LINKSRC = fpgetround.3c
```



```
1646 fpsetround.3c                     := LINKSRC = fpgetround.3c
1647 fpsetsticky.3c                    := LINKSRC = fpgetround.3c

1649 putc.3c                           := LINKSRC = fputc.3c
1650 putc_unlocked.3c                  := LINKSRC = fputc.3c
1651 putchar.3c                        := LINKSRC = fputc.3c
1652 putchar_unlocked.3c               := LINKSRC = fputc.3c
1653 putw.3c                           := LINKSRC = fputc.3c

1655 putwc.3c                          := LINKSRC = fputwc.3c
1656 putwchar.3c                       := LINKSRC = fputwc.3c

1658 fseeko.3c                         := LINKSRC = fseek.3c

1660 ftello.3c                         := LINKSRC = ftell.3c

1662 nftw.3c                           := LINKSRC = ftw.3c

1664 swprintf.3c                       := LINKSRC = fwprintf.3c
1665 wprintf.3c                        := LINKSRC = fwprintf.3c

1667 swscanf.3c                        := LINKSRC = fwscanf.3c
1668 vfwscanf.3c                       := LINKSRC = fwscanf.3c
1669 vswscanf.3c                       := LINKSRC = fwscanf.3c
1670 vwscanf.3c                        := LINKSRC = fwscanf.3c
1671 wscanf.3c                         := LINKSRC = fwscanf.3c

1673 gethomelgroup.3c                  := LINKSRC = getcpuid.3c

1675 endgrent.3c                       := LINKSRC = getgrnam.3c
1676 fgetgrent.3c                      := LINKSRC = getgrnam.3c
1677 fgetgrent_r.3c                    := LINKSRC = getgrnam.3c
1678 getgrent.3c                       := LINKSRC = getgrnam.3c
1679 getgrent_r.3c                     := LINKSRC = getgrnam.3c
1680 getgrgid.3c                       := LINKSRC = getgrnam.3c
1681 getgrgid_r.3c                     := LINKSRC = getgrnam.3c
1682 getgrnam_r.3c                     := LINKSRC = getgrnam.3c
1683 setgrent.3c                       := LINKSRC = getgrnam.3c

1685 sethostname.3c                    := LINKSRC = gethostname.3c

1687 gethrvtime.3c                     := LINKSRC = gethrtime.3c

1689 getdelim.3c                       := LINKSRC = getline.3c

1691 getlogin_r.3c                     := LINKSRC = getlogin.3c

1693 getextmntent.3c                   := LINKSRC = getmntent.3c
1694 getmntany.3c                     := LINKSRC = getmntent.3c
1695 hasmntopt.3c                     := LINKSRC = getmntent.3c
1696 putmntent.3c                     := LINKSRC = getmntent.3c
1697 resetmnttab.3c                   := LINKSRC = getmntent.3c

1699 endnetgrent.3c                   := LINKSRC = getnetgrent.3c
1700 getnetgrent_r.3c                 := LINKSRC = getnetgrent.3c
1701 innetgr.3c                       := LINKSRC = getnetgrent.3c
1702 setnetgrent.3c                   := LINKSRC = getnetgrent.3c

1704 get_nprocs_conf.3c               := LINKSRC = get_nprocs.3c

1706 getpassphrase.3c                 := LINKSRC = getpass.3c

1708 setpriority.3c                   := LINKSRC = getpriority.3c

1710 setprogname.3c                   := LINKSRC = getprogname.3c
```

```
endpwent.3c                             := LINKSRC = getpwnam.3c
fgetpwent.3c                            := LINKSRC = getpwnam.3c
fgetpwent_r.3c                          := LINKSRC = getpwnam.3c
getpwent.3c                             := LINKSRC = getpwnam.3c
getpwent_r.3c                           := LINKSRC = getpwnam.3c
getpwnam_r.3c                           := LINKSRC = getpwnam.3c
getpwuid.3c                             := LINKSRC = getpwnam.3c
getpwuid_r.3c                           := LINKSRC = getpwnam.3c
setpwent.3c                             := LINKSRC = getpwnam.3c

fgets.3c                                := LINKSRC = gets.3c

endspent.3c                             := LINKSRC = getspnam.3c
fgetspent.3c                            := LINKSRC = getspnam.3c
fgetspent_r.3c                          := LINKSRC = getspnam.3c
getspent.3c                             := LINKSRC = getspnam.3c
getspent_r.3c                           := LINKSRC = getspnam.3c
getspnam_r.3c                           := LINKSRC = getspnam.3c
setspent.3c                             := LINKSRC = getspnam.3c

bind_textdomain_codeset.3c              := LINKSRC = gettext.3c
bindtextdomain.3c                       := LINKSRC = gettext.3c
dcgettext.3c                            := LINKSRC = gettext.3c
dcngettext.3c                           := LINKSRC = gettext.3c
dgettext.3c                             := LINKSRC = gettext.3c
dngettext.3c                            := LINKSRC = gettext.3c
ngettext.3c                             := LINKSRC = gettext.3c
textdomain.3c                           := LINKSRC = gettext.3c

settimeofday.3c                         := LINKSRC = gettimeofday.3c

endusershell.3c                         := LINKSRC = getusershell.3c
setusershell.3c                         := LINKSRC = getusershell.3c

endutent.3c                             := LINKSRC = getutent.3c
getutid.3c                              := LINKSRC = getutent.3c
getutline.3c                            := LINKSRC = getutent.3c
pututline.3c                            := LINKSRC = getutent.3c
setutent.3c                             := LINKSRC = getutent.3c
utmpname.3c                             := LINKSRC = getutent.3c

endutxent.3c                            := LINKSRC = getutxent.3c
getutmp.3c                              := LINKSRC = getutxent.3c
getutmpx.3c                             := LINKSRC = getutxent.3c
getutxid.3c                             := LINKSRC = getutxent.3c
getutxline.3c                           := LINKSRC = getutxent.3c
pututxline.3c                           := LINKSRC = getutxent.3c
setutxent.3c                            := LINKSRC = getutxent.3c
updwtmp.3c                              := LINKSRC = getutxent.3c
updwtmpx.3c                             := LINKSRC = getutxent.3c
utmpxname.3c                            := LINKSRC = getutxent.3c

getvfsany.3c                            := LINKSRC = getvfsent.3c
getvfsfile.3c                           := LINKSRC = getvfsent.3c
getvfsspec.3c                           := LINKSRC = getvfsent.3c

getwc_l.3c                              := LINKSRC = getwc.3c

getwchar_l.3c                           := LINKSRC = getwchar.3c

fgetws.3c                               := LINKSRC = getws.3c

getzoneidbyname.3c                      := LINKSRC = getzoneid.3c
getzonenamebyid.3c                      := LINKSRC = getzoneid.3c

globfree.3c                             := LINKSRC = glob.3c
```



```
hcreate.3c                              := LINKSRC = hsearch.3c
hdestroy.3c                             := LINKSRC = hsearch.3c

rindex.3c                               := LINKSRC = index.3c

remque.3c                               := LINKSRC = insque.3c

finite.3c                               := LINKSRC = isnand.3c
fpclass.3c                              := LINKSRC = isnand.3c
isnanf.3c                               := LINKSRC = isnand.3c
unordered.3c                            := LINKSRC = isnand.3c

isenglish.3c                            := LINKSRC = iswalpha.3c
isideogram.3c                           := LINKSRC = iswalpha.3c
isnumber.3c                             := LINKSRC = iswalpha.3c
isphonogram.3c                          := LINKSRC = iswalpha.3c
isspecial.3c                            := LINKSRC = iswalpha.3c
iswalnum.3c                             := LINKSRC = iswalpha.3c
iswalnum_l.3c                           := LINKSRC = iswalpha.3c
iswalpha_l.3c                           := LINKSRC = iswalpha.3c
iswascii.3c                             := LINKSRC = iswalpha.3c
iswblank.3c                             := LINKSRC = iswalpha.3c
iswblank_l.3c                           := LINKSRC = iswalpha.3c
iswcntrl.3c                             := LINKSRC = iswalpha.3c
iswcntrl_l.3c                           := LINKSRC = iswalpha.3c
iswdigit.3c                             := LINKSRC = iswalpha.3c
iswdigit_l.3c                           := LINKSRC = iswalpha.3c
iswgraph.3c                             := LINKSRC = iswalpha.3c
iswgraph_l.3c                           := LINKSRC = iswalpha.3c
iswhexnumber.3c                         := LINKSRC = iswalpha.3c
iswhexnumber_l.3c                       := LINKSRC = iswalpha.3c
iswideogram.3c                          := LINKSRC = iswalpha.3c
iswideogram_l.3c                        := LINKSRC = iswalpha.3c
iswlower.3c                             := LINKSRC = iswalpha.3c
iswlower_l.3c                           := LINKSRC = iswalpha.3c
iswnumber.3c                            := LINKSRC = iswalpha.3c
iswnumber_l.3c                          := LINKSRC = iswalpha.3c
iswphonogram.3c                         := LINKSRC = iswalpha.3c
iswphonogram_l.3c                       := LINKSRC = iswalpha.3c
iswprint.3c                             := LINKSRC = iswalpha.3c
iswprint_l.3c                           := LINKSRC = iswalpha.3c
iswpunct.3c                             := LINKSRC = iswalpha.3c
iswpunct_l.3c                           := LINKSRC = iswalpha.3c
iswspace.3c                             := LINKSRC = iswalpha.3c
iswspace_l.3c                           := LINKSRC = iswalpha.3c
iswspecial.3c                           := LINKSRC = iswalpha.3c
iswspecial_l.3c                         := LINKSRC = iswalpha.3c
iswupper.3c                             := LINKSRC = iswalpha.3c
iswupper_l.3c                           := LINKSRC = iswalpha.3c
iswxdigit.3c                            := LINKSRC = iswalpha.3c
iswxdigit_l.3c                          := LINKSRC = iswalpha.3c

iswctype_l.3c                           := LINKSRC = iswctype.3c

ulckpwdf.3c                             := LINKSRC = lckpwdf.3c

lfind.3c                                := LINKSRC = lsearch.3c

swapcontext.3c                          := LINKSRC = makecontext.3c

major.3c                                := LINKSRC = makedev.3c
minor.3c                                := LINKSRC = makedev.3c

alloca.3c                               := LINKSRC = malloc.3c
calloc.3c                               := LINKSRC = malloc.3c
```

```
free.3c                                 := LINKSRC = malloc.3c
memalign.3c                             := LINKSRC = malloc.3c
realloc.3c                              := LINKSRC = malloc.3c
valloc.3c                               := LINKSRC = malloc.3c

mblen_l.3c                              := LINKSRC = mblen.3c

mbrlen_l.3c                             := LINKSRC = mbrlen.3c

mbrtowc_l.3c                            := LINKSRC = mbrtowc.3c

mbsinit_l.3c                            := LINKSRC = mbsinit.3c

mbsnrtowcs.3c                           := LINKSRC = mbsrtowcs.3c
mbsnrtowcs_l.3c                         := LINKSRC = mbsrtowcs.3c
mbsrtowcs_l.3c                          := LINKSRC = mbsrtowcs.3c
mbstowcs.3c                             := LINKSRC = mbsrtowcs.3c
mbstowcs_l.3c                           := LINKSRC = mbsrtowcs.3c

mbtowc_l.3c                             := LINKSRC = mbtowc.3c

membar_consumer.3c                      := LINKSRC = membar_ops.3c
membar_enter.3c                         := LINKSRC = membar_ops.3c
membar_exit.3c                          := LINKSRC = membar_ops.3c
membar_producer.3c                      := LINKSRC = membar_ops.3c

memccpy.3c                              := LINKSRC = memory.3c
memchr.3c                               := LINKSRC = memory.3c
memcmp.3c                               := LINKSRC = memory.3c
memcpy.3c                               := LINKSRC = memory.3c
memmem.3c                               := LINKSRC = memory.3c
memmove.3c                              := LINKSRC = memory.3c
memset.3c                               := LINKSRC = memory.3c

mkfifoat.3c                             := LINKSRC = mkfifo.3c

mkdtemp.3c                              := LINKSRC = mkstemp.3c
mkostemp.3c                             := LINKSRC = mkstemp.3c
mkostemps.3c                            := LINKSRC = mkstemp.3c
mkstemps.3c                             := LINKSRC = mkstemp.3c

munlock.3c                              := LINKSRC = mlock.3c

munlockall.3c                           := LINKSRC = mlockall.3c

mq_reltimedreceive_np.3c                := LINKSRC = mq_receive.3c
mq_timedreceive.3c                      := LINKSRC = mq_receive.3c

mq_reltimedsend_np.3c                   := LINKSRC = mq_send.3c
mq_timedsend.3c                         := LINKSRC = mq_send.3c

mutex_consistent.3c                     := LINKSRC = mutex_init.3c
mutex_destroy.3c                        := LINKSRC = mutex_init.3c
mutex_lock.3c                           := LINKSRC = mutex_init.3c
mutex_trylock.3c                        := LINKSRC = mutex_init.3c
mutex_unlock.3c                         := LINKSRC = mutex_init.3c

dbm_clearerr.3c                         := LINKSRC = ndbm.3c
dbm_close.3c                            := LINKSRC = ndbm.3c
dbm_delete.3c                           := LINKSRC = ndbm.3c
dbm_error.3c                            := LINKSRC = ndbm.3c
dbm_fetch.3c                            := LINKSRC = ndbm.3c
dbm_firstkey.3c                         := LINKSRC = ndbm.3c
dbm_nextkey.3c                          := LINKSRC = ndbm.3c
dbm_open.3c                             := LINKSRC = ndbm.3c
dbm_store.3c                            := LINKSRC = ndbm.3c
```



```
duplocale.3c                            := LINKSRC = newlocale.3c
freelocale.3c                           := LINKSRC = newlocale.3c

nl_langinfo_l.3c                        := LINKSRC = nl_langinfo.3c

fdopendir.3c                            := LINKSRC = opendir.3c

errno.3c                                := LINKSRC = perror.3c

pclose.3c                               := LINKSRC = popen.3c

port_dissociate.3c                      := LINKSRC = port_associate.3c

port_getn.3c                            := LINKSRC = port_get.3c

port_sendn.3c                           := LINKSRC = port_send.3c

posix_spawnp.3c                         := LINKSRC = posix_spawn.3c

posix_spawn_file_actions_addopen.3c     := LINKSRC = posix_spawn_file_actions_ad
posix_spawn_file_actions_init.3c        := LINKSRC = posix_spawn_file_actions_de

posix_spawnattr_init.3c                 := LINKSRC = posix_spawnattr_destroy.3c

posix_spawnattr_setflags.3c             := LINKSRC = posix_spawnattr_getflags.3c

posix_spawnattr_setpgroup.3c            := LINKSRC = posix_spawnattr_getpgroup.3

posix_spawnattr_setschedparam.3c        := LINKSRC = posix_spawnattr_getschedpar

posix_spawnattr_setschedpolicy.3c       := LINKSRC = posix_spawnattr_getschedpol

posix_spawnattr_setsigdefault.3c        := LINKSRC = posix_spawnattr_getsigdefau

posix_spawnattr_setsigignore_np.3c      := LINKSRC = posix_spawnattr_getsigignor

posix_spawnattr_setsigmask.3c           := LINKSRC = posix_spawnattr_getsigmask.

asprintf.3c                             := LINKSRC = printf.3c
fprintf.3c                              := LINKSRC = printf.3c
snprintf.3c                             := LINKSRC = printf.3c
sprintf.3c                              := LINKSRC = printf.3c

priv_allocset.3c                        := LINKSRC = priv_addset.3c
priv_basicset.3c                        := LINKSRC = priv_addset.3c
priv_copyset.3c                         := LINKSRC = priv_addset.3c
priv_defaultset.3c                      := LINKSRC = priv_addset.3c
#endif /* ! codereview */
priv_delset.3c                          := LINKSRC = priv_addset.3c
priv_emptyset.3c                        := LINKSRC = priv_addset.3c
priv_fillset.3c                         := LINKSRC = priv_addset.3c
priv_freeset.3c                         := LINKSRC = priv_addset.3c
priv_intersect.3c                       := LINKSRC = priv_addset.3c
priv_inverse.3c                         := LINKSRC = priv_addset.3c
priv_isemptyset.3c                      := LINKSRC = priv_addset.3c
priv_isequalset.3c                      := LINKSRC = priv_addset.3c
priv_isfullset.3c                       := LINKSRC = priv_addset.3c
priv_ismember.3c                        := LINKSRC = priv_addset.3c
priv_issubset.3c                        := LINKSRC = priv_addset.3c
priv_union.3c                           := LINKSRC = priv_addset.3c

priv_ineffect.3c                        := LINKSRC = priv_set.3c

priv_getbyname.3c                       := LINKSRC = priv_str_to_set.3c
priv_getbynum.3c                        := LINKSRC = priv_str_to_set.3c
```

```
priv_getsetbyname.3c                    := LINKSRC = priv_str_to_set.3c
priv_getsetbynum.3c                     := LINKSRC = priv_str_to_set.3c
priv_gettext.3c                         := LINKSRC = priv_str_to_set.3c
priv_set_to_str.3c                      := LINKSRC = priv_str_to_set.3c

psiginfo.3c                             := LINKSRC = psignal.3c

pthread_attr_setdetachstate.3c          := LINKSRC = pthread_attr_getdetachstate

pthread_attr_setguardsize.3c            := LINKSRC = pthread_attr_getguardsize.3

pthread_attr_setinheritsched.3c         := LINKSRC = pthread_attr_getinheritsche

pthread_attr_setschedparam.3c           := LINKSRC = pthread_attr_getschedparam.

pthread_attr_setschedpolicy.3c          := LINKSRC = pthread_attr_getschedpolicy

pthread_attr_setscope.3c                := LINKSRC = pthread_attr_getscope.3c
pthread_attr_setstack.3c                := LINKSRC = pthread_attr_getstack.3c

pthread_attr_setstackaddr.3c            := LINKSRC = pthread_attr_getstackaddr.3

pthread_attr_setstacksize.3c            := LINKSRC = pthread_attr_getstacksize.3

pthread_attr_destroy.3c                 := LINKSRC = pthread_attr_init.3c

pthread_barrier_init.3c                 := LINKSRC = pthread_barrier_destroy.3c

pthread_barrierattr_init.3c             := LINKSRC = pthread_barrierattr_destroy

pthread_barrierattr_setpshared.3c       := LINKSRC = pthread_barrierattr_getpsha

pthread_cond_destroy.3c                 := LINKSRC = pthread_cond_init.3c

pthread_cond_broadcast.3c               := LINKSRC = pthread_cond_signal.3c

pthread_cond_reltimedwait_np.3c         := LINKSRC = pthread_cond_wait.3c
pthread_cond_timedwait.3c               := LINKSRC = pthread_cond_wait.3c

pthread_condattr_setclock.3c            := LINKSRC = pthread_condattr_getclock.3

pthread_condattr_setpshared.3c          := LINKSRC = pthread_condattr_getpshared

pthread_condattr_destroy.3c             := LINKSRC = pthread_condattr_init.3c

pthread_setconcurrency.3c               := LINKSRC = pthread_getconcurrency.3c

pthread_setschedparam.3c                := LINKSRC = pthread_getschedparam.3c

pthread_setspecific.3c                  := LINKSRC = pthread_getspecific.3c

pthread_key_create_once_np.3c           := LINKSRC = pthread_key_create.3c

pthread_mutex_setprioceiling.3c         := LINKSRC = pthread_mutex_getprioceilin

pthread_mutex_destroy.3c                := LINKSRC = pthread_mutex_init.3c

pthread_mutex_trylock.3c                := LINKSRC = pthread_mutex_lock.3c
pthread_mutex_unlock.3c                 := LINKSRC = pthread_mutex_lock.3c

pthread_mutex_reltimedlock_np.3c        := LINKSRC = pthread_mutex_timedlock.3c

pthread_mutexattr_setprioceiling.3c     := LINKSRC = pthread_mutexattr_getprioce

pthread_mutexattr_setprotocol.3c        := LINKSRC = pthread_mutexattr_getprotoc
```



```
pthread_mutexattr_setpshared.3c         := LINKSRC = pthread_mutexattr_getpshare

pthread_mutexattr_setrobust.3c          := LINKSRC = pthread_mutexattr_getrobust

pthread_mutexattr_settype.3c            := LINKSRC = pthread_mutexattr_gettype.3

pthread_mutexattr_destroy.3c            := LINKSRC = pthread_mutexattr_init.3c

pthread_rwlock_destroy.3c               := LINKSRC = pthread_rwlock_init.3c

pthread_rwlock_tryrdlock.3c             := LINKSRC = pthread_rwlock_rdlock.3c

pthread_rwlock_reltimedrdlock_np.3c     := LINKSRC = pthread_rwlock_timedrdlock.

pthread_rwlock_reltimedwrlock_np.3c     := LINKSRC = pthread_rwlock_timedwrlock.

pthread_rwlock_trywrlock.3c             := LINKSRC = pthread_rwlock_wrlock.3c

pthread_rwlockattr_setpshared.3c        := LINKSRC = pthread_rwlockattr_getpshar

pthread_rwlockattr_destroy.3c           := LINKSRC = pthread_rwlockattr_init.3c

pthread_spin_init.3c                    := LINKSRC = pthread_spin_destroy.3c

pthread_spin_trylock.3c                 := LINKSRC = pthread_spin_lock.3c

fputs.3c                                := LINKSRC = puts.3c

rand_r.3c                               := LINKSRC = rand.3c
srand.3c                                := LINKSRC = rand.3c

initstate.3c                            := LINKSRC = random.3c
setstate.3c                             := LINKSRC = random.3c
srandom.3c                              := LINKSRC = random.3c

rctlblk_get_enforced_value.3c           := LINKSRC = rctlblk_set_value.3c
rctlblk_get_firing_time.3c              := LINKSRC = rctlblk_set_value.3c
rctlblk_get_global_action.3c            := LINKSRC = rctlblk_set_value.3c
rctlblk_get_global_flags.3c             := LINKSRC = rctlblk_set_value.3c
rctlblk_get_local_action.3c             := LINKSRC = rctlblk_set_value.3c
rctlblk_get_local_flags.3c              := LINKSRC = rctlblk_set_value.3c
rctlblk_get_privilege.3c                := LINKSRC = rctlblk_set_value.3c
rctlblk_get_recipient_pid.3c            := LINKSRC = rctlblk_set_value.3c
rctlblk_get_value.3c                    := LINKSRC = rctlblk_set_value.3c
rctlblk_set_local_action.3c             := LINKSRC = rctlblk_set_value.3c
rctlblk_set_local_flags.3c              := LINKSRC = rctlblk_set_value.3c
rctlblk_set_privilege.3c                := LINKSRC = rctlblk_set_value.3c
rctlblk_set_recipient_pid.3c            := LINKSRC = rctlblk_set_value.3c
rctlblk_size.3c                         := LINKSRC = rctlblk_set_value.3c

re_exec.3c                              := LINKSRC = re_comp.3c

readdir_r.3c                            := LINKSRC = readdir.3c

regex.3c                                := LINKSRC = regcmp.3c

regerror.3c                             := LINKSRC = regcomp.3c
regexec.3c                              := LINKSRC = regcomp.3c
regfree.3c                              := LINKSRC = regcomp.3c

rw_rdlock.3c                            := LINKSRC = rwlock.3c
rw_tryrdlock.3c                         := LINKSRC = rwlock.3c
rw_trywrlock.3c                         := LINKSRC = rwlock.3c
rw_unlock.3c                            := LINKSRC = rwlock.3c
rw_wrlock.3c                            := LINKSRC = rwlock.3c
rwlock_destroy.3c                       := LINKSRC = rwlock.3c
```

```
2108 rwlock_init.3c                          := LINKSRC = rwlock.3c

2110 alphasort.3c                            := LINKSRC = scandir.3c

2112 fscanf.3c                               := LINKSRC = scanf.3c
2113 sscanf.3c                               := LINKSRC = scanf.3c
2114 vfscanf.3c                              := LINKSRC = scanf.3c
2115 vscanf.3c                               := LINKSRC = scanf.3c
2116 vsscanf.3c                              := LINKSRC = scanf.3c

2118 sched_get_priority_min.3c               := LINKSRC = sched_get_priority_max.3c

2120 schedctl_exit.3c                        := LINKSRC = schedctl_init.3c
2121 schedctl_lookup.3c                      := LINKSRC = schedctl_init.3c
2122 schedctl_start.3c                       := LINKSRC = schedctl_init.3c
2123 schedctl_stop.3c                        := LINKSRC = schedctl_init.3c

2125 FD_CLR.3c                               := LINKSRC = select.3c
2126 FD_ISSET.3c                             := LINKSRC = select.3c
2127 FD_SET.3c                               := LINKSRC = select.3c
2128 FD_ZERO.3c                              := LINKSRC = select.3c
2129 pselect.3c                              := LINKSRC = select.3c

2131 sem_reltimedwait_np.3c                  := LINKSRC = sem_timedwait.3c

2133 sem_trywait.3c                          := LINKSRC = sem_wait.3c

2135 sema_destroy.3c                         := LINKSRC = semaphore.3c
2136 sema_init.3c                            := LINKSRC = semaphore.3c
2137 sema_post.3c                            := LINKSRC = semaphore.3c
2138 sema_trywait.3c                         := LINKSRC = semaphore.3c
2139 sema_wait.3c                            := LINKSRC = semaphore.3c

2141 setvbuf.3c                              := LINKSRC = setbuf.3c

2143 setlinebuf.3c                           := LINKSRC = setbuffer.3c

2145 longjmp.3c                              := LINKSRC = setjmp.3c
2146 siglongjmp.3c                           := LINKSRC = setjmp.3c
2147 sigsetjmp.3c                           := LINKSRC = setjmp.3c

2149 sighold.3c                             := LINKSRC = signal.3c
2150 sigignore.3c                           := LINKSRC = signal.3c
2151 sigpause.3c                            := LINKSRC = signal.3c
2152 sigrelse.3c                            := LINKSRC = signal.3c
2153 sigset.3c                              := LINKSRC = signal.3c

2155 sigaddset.3c                           := LINKSRC = sigsetops.3c
2156 sigdelset.3c                           := LINKSRC = sigsetops.3c
2157 sigemptyset.3c                         := LINKSRC = sigsetops.3c
2158 sigfillset.3c                          := LINKSRC = sigsetops.3c
2159 sigismember.3c                        := LINKSRC = sigsetops.3c

2161 sigtimedwait.3c                        := LINKSRC = sigwaitinfo.3c

2163 gsignal.3c                             := LINKSRC = ssignal.3c

2165 stderr.3c                              := LINKSRC = stdio.3c
2166 stdin.3c                               := LINKSRC = stdio.3c
2167 stdout.3c                              := LINKSRC = stdio.3c

2169 sig2str.3c                             := LINKSRC = str2sig.3c

2171 strcoll_l.3c                           := LINKSRC = strcoll.3c

2173 strerror_l.3c                          := LINKSRC = strerror.3c
```



```
2174 strerror_r.3c                          := LINKSRC = strerror.3c

2176 strfmon_l.3c                           := LINKSRC = strfmon.3c

2178 ascftime.3c                            := LINKSRC = strftime.3c
2179 cftime.3c                              := LINKSRC = strftime.3c
2180 strftime_l.3c                          := LINKSRC = strftime.3c

2182 stpcpy.3c                              := LINKSRC = string.3c
2183 stpncpy.3c                             := LINKSRC = string.3c
2184 strcasecmp.3c                          := LINKSRC = string.3c
2185 strcasecmp_l.3c                        := LINKSRC = string.3c
2186 strcasestr.3c                          := LINKSRC = string.3c
2187 strcasestr_l.3c                        := LINKSRC = string.3c
2188 strcat.3c                              := LINKSRC = string.3c
2189 strchr.3c                              := LINKSRC = string.3c
2190 strchrnul.3c                           := LINKSRC = string.3c
2191 strcmp.3c                              := LINKSRC = string.3c
2192 strcpy.3c                              := LINKSRC = string.3c
2193 strcspn.3c                             := LINKSRC = string.3c
2194 strdup.3c                              := LINKSRC = string.3c
2195 strdupa.3c                             := LINKSRC = string.3c
2196 strlcat.3c                             := LINKSRC = string.3c
2197 strlcpy.3c                             := LINKSRC = string.3c
2198 strlen.3c                              := LINKSRC = string.3c
2199 strncasecmp.3c                         := LINKSRC = string.3c
2200 strncasecmp_l.3c                       := LINKSRC = string.3c
2201 strncat.3c                             := LINKSRC = string.3c
2202 strncmp.3c                             := LINKSRC = string.3c
2203 strncpy.3c                             := LINKSRC = string.3c
2204 strndup.3c                             := LINKSRC = string.3c
2205 strndupa.3c                            := LINKSRC = string.3c
2206 strnlen.3c                             := LINKSRC = string.3c
2207 strnstr.3c                             := LINKSRC = string.3c
2208 strpbrk.3c                             := LINKSRC = string.3c
2209 strrchr.3c                             := LINKSRC = string.3c
2210 strsep.3c                              := LINKSRC = string.3c
2211 strspn.3c                              := LINKSRC = string.3c
2212 strstr.3c                              := LINKSRC = string.3c
2213 strtok.3c                              := LINKSRC = string.3c
2214 strtok_r.3c                            := LINKSRC = string.3c

2216 file_to_decimal.3c                     := LINKSRC = string_to_decimal.3c
2217 func_to_decimal.3c                     := LINKSRC = string_to_decimal.3c

2219 strptime_l.3c                          := LINKSRC = strptime.3c

2221 atof.3c                               := LINKSRC = strtod.3c
2222 strtof.3c                              := LINKSRC = strtod.3c
2223 strtold.3c                             := LINKSRC = strtod.3c

2225 strtoumax.3c                          := LINKSRC = strtoimax.3c

2227 atoi.3c                               := LINKSRC = strtol.3c
2228 atol.3c                               := LINKSRC = strtol.3c
2229 atoll.3c                              := LINKSRC = strtol.3c
2230 lltostr.3c                            := LINKSRC = strtol.3c
2231 strtoll.3c                            := LINKSRC = strtol.3c
2232 ulltostr.3c                           := LINKSRC = strtol.3c

2234 strtoull.3c                           := LINKSRC = strtoul.3c

2236 wstostr.3c                            := LINKSRC = strtows.3c

2238 strxfrm_l.3c                          := LINKSRC = strxfrm.3c
```

```
2240 closelog.3c                        := LINKSRC = syslog.3c
2241 openlog.3c                         := LINKSRC = syslog.3c
2242 setlogmask.3c                      := LINKSRC = syslog.3c

2244 thr_setconcurrency.3c              := LINKSRC = thr_getconcurrency.3c

2246 thr_setprio.3c                     := LINKSRC = thr_getprio.3c

2248 thr_getspecific.3c                 := LINKSRC = thr_keycreate.3c
2249 thr_keycreate_once.3c             := LINKSRC = thr_keycreate.3c
2250 thr_setspecific.3c                := LINKSRC = thr_keycreate.3c

2252 thr_continue.3c                    := LINKSRC = thr_suspend.3c

2254 timer_getoverrun.3c                := LINKSRC = timer_settime.3c
2255 timer_gettime.3c                   := LINKSRC = timer_settime.3c

2257 timerclear.3c                      := LINKSRC = timeradd.3c
2258 timercmp.3c                        := LINKSRC = timeradd.3c
2259 timerisset.3c                      := LINKSRC = timeradd.3c
2260 timersub.3c                        := LINKSRC = timeradd.3c

2262 timerfd_gettime.3c                        := LINKSRC = timerfd_create.3c
2263 timerfd_settime.3c                        := LINKSRC = timerfd_create.3c

2265 tempnam.3c                         := LINKSRC = tmpnam.3c
2266 tmpnam_r.3c                        := LINKSRC = tmpnam.3c

2268 tolower_l.3c                       := LINKSRC = tolower.3c

2270 toupper_l.3c                       := LINKSRC = toupper.3c

2272 towlower_l.3c                      := LINKSRC = towlower.3c

2274 towupper_l.3c                      := LINKSRC = towupper.3c

2276 ftruncate.3c                       := LINKSRC = truncate.3c

2278 tdelete.3c                         := LINKSRC = tsearch.3c
2279 tfind.3c                           := LINKSRC = tsearch.3c
2280 twalk.3c                           := LINKSRC = tsearch.3c

2282 ttyname_r.3c                       := LINKSRC = ttyname.3c

2284 uconv_u16tou8.3c                   := LINKSRC = uconv_u16tou32.3c
2285 uconv_u32tou16.3c                  := LINKSRC = uconv_u16tou32.3c
2286 uconv_u32tou8.3c                   := LINKSRC = uconv_u16tou32.3c
2287 uconv_u8tou16.3c                   := LINKSRC = uconv_u16tou32.3c
2288 uconv_u8tou32.3c                   := LINKSRC = uconv_u16tou32.3c

2290 ucred_free.3c                      := LINKSRC = ucred.3c
2291 ucred_get.3c                       := LINKSRC = ucred.3c
2292 ucred_getegid.3c                   := LINKSRC = ucred.3c
2293 ucred_geteuid.3c                   := LINKSRC = ucred.3c
2294 ucred_getgroups.3c                 := LINKSRC = ucred.3c
2295 ucred_getlabel.3c                  := LINKSRC = ucred.3c
2296 ucred_getpflags.3c                 := LINKSRC = ucred.3c
2297 ucred_getpid.3c                    := LINKSRC = ucred.3c
2298 ucred_getprivset.3c                := LINKSRC = ucred.3c
2299 ucred_getprojid.3c                 := LINKSRC = ucred.3c
2300 ucred_getrgid.3c                   := LINKSRC = ucred.3c
2301 ucred_getruid.3c                   := LINKSRC = ucred.3c
2302 ucred_getsgid.3c                   := LINKSRC = ucred.3c
2303 ucred_getsuid.3c                   := LINKSRC = ucred.3c
2304 ucred_getzoneid.3c                 := LINKSRC = ucred.3c
2305 ucred_size.3c                      := LINKSRC = ucred.3c
```



```
2307 vswprintf.3c                       := LINKSRC = vfwprintf.3c
2308 vwprintf.3c                        := LINKSRC = vfwprintf.3c

2310 vasprintf.3c                       := LINKSRC = vprintf.3c
2311 vfprintf.3c                        := LINKSRC = vprintf.3c
2312 vsnprintf.3c                       := LINKSRC = vprintf.3c
2313 vsprintf.3c                        := LINKSRC = vprintf.3c

2315 wait4.3c                           := LINKSRC = wait3.3c

2317 addrtosymstr.3c                    := LINKSRC = walkcontext.3c
2318 backtrace.3c                       := LINKSRC = walkcontext.3c
2319 backtrace_symbols.3c               := LINKSRC = walkcontext.3c
2320 backtrace_symbols_fd.3c            := LINKSRC = walkcontext.3c
2321 printstack.3c                      := LINKSRC = walkcontext.3c

2323 wcpncpy.3c                        := LINKSRC = wcpcpy.3c

2325 wcrtomb_l.3c                       := LINKSRC = wcrtomb.3c

2327 wcscasecmp_l.3c                    := LINKSRC = wcscasecmp.3c
2328 wcsncasecmp.3c                     := LINKSRC = wcscasecmp.3c
2329 wcsncasecmp_l.3c                   := LINKSRC = wcscasecmp.3c

2331 wcscoll_l.3c                       := LINKSRC = wcscoll.3c
2332 wscoll.3c                          := LINKSRC = wcscoll.3c

2334 wcsnlen.3c                         := LINKSRC = wcslen.3c

2336 wcsnrtombs.3c                      := LINKSRC = wcsrtombs.3c
2337 wcsnrtombs_l.3c                    := LINKSRC = wcsrtombs.3c
2338 wcsrtombs_l.3c                     := LINKSRC = wcsrtombs.3c

2340 watof.3c                          := LINKSRC = wcstod.3c
2341 wcstof.3c                          := LINKSRC = wcstod.3c
2342 wcstold.3c                         := LINKSRC = wcstod.3c
2343 wstod.3c                           := LINKSRC = wcstod.3c

2345 wcstoumax.3c                       := LINKSRC = wcstoimax.3c

2347 watoi.3c                          := LINKSRC = wcstol.3c
2348 watol.3c                          := LINKSRC = wcstol.3c
2349 watoll.3c                         := LINKSRC = wcstol.3c
2350 wcstoll.3c                        := LINKSRC = wcstol.3c
2351 wstol.3c                          := LINKSRC = wcstol.3c

2353 wcstoull.3c                       := LINKSRC = wcstoul.3c

2355 wcscat.3c                         := LINKSRC = wcstring.3c
2356 wcschr.3c                         := LINKSRC = wcstring.3c
2357 wcscmp.3c                         := LINKSRC = wcstring.3c
2358 wcscpy.3c                         := LINKSRC = wcstring.3c
2359 wcscspn.3c                        := LINKSRC = wcstring.3c
2360 wcsncat.3c                        := LINKSRC = wcstring.3c
2361 wcsncmp.3c                        := LINKSRC = wcstring.3c
2362 wcsncpy.3c                        := LINKSRC = wcstring.3c
2363 wcspbrk.3c                        := LINKSRC = wcstring.3c
2364 wcsrchr.3c                        := LINKSRC = wcstring.3c
2365 wcsspn.3c                         := LINKSRC = wcstring.3c
2366 wcstok.3c                         := LINKSRC = wcstring.3c
2367 wcswcs.3c                         := LINKSRC = wcstring.3c
2368 windex.3c                         := LINKSRC = wcstring.3c
2369 wrindex.3c                        := LINKSRC = wcstring.3c
2370 wscat.3c                          := LINKSRC = wcstring.3c
2371 wschr.3c                          := LINKSRC = wcstring.3c
```

```
2372 wscmp.3c                                := LINKSRC = wcstring.3c
2373 wscpy.3c                                := LINKSRC = wcstring.3c
2374 wscspn.3c                               := LINKSRC = wcstring.3c
2375 wslen.3c                                := LINKSRC = wcstring.3c
2376 wsncat.3c                               := LINKSRC = wcstring.3c
2377 wsncmp.3c                               := LINKSRC = wcstring.3c
2378 wsncpy.3c                               := LINKSRC = wcstring.3c
2379 wspbrk.3c                               := LINKSRC = wcstring.3c
2380 wsrchr.3c                               := LINKSRC = wcstring.3c
2381 wsspn.3c                                := LINKSRC = wcstring.3c
2382 wstok.3c                                := LINKSRC = wcstring.3c
2383 wscasecmp.3c                            := LINKSRC = wstring.3c
2384 wscol.3c                                := LINKSRC = wstring.3c
2385 wsdup.3c                                := LINKSRC = wstring.3c
2386 wsncasecmp.3c                           := LINKSRC = wstring.3c

2388 wcswidth_l.3c                           := LINKSRC = wcswidth.3c

2390 wsxfrm.3c                               := LINKSRC = wcsxfrm.3c

2392 wctob_l.3c                             := LINKSRC = wctob.3c

2394 wctomb_l.3c                            := LINKSRC = wctomb.3c

2396 towctrans.3c                           := LINKSRC = wctrans.3c
2397 towctrans_l.3c                         := LINKSRC = wctrans.3c
2398 wctrans_l.3c                           := LINKSRC = wctrans.3c

2400 wctype_l.3c                            := LINKSRC = wctype.3c

2402 wcwidth_l.3c                           := LINKSRC = wcwidth.3c

2404 wordfree.3c                            := LINKSRC = wordexp.3c

2406 .KEEP_STATE:

2408 include         $(SRC)/man/Makefile.man

2410 install:        $(ROOTMANFILES) $(ROOTMANLINKS)
```

```
**********************************************************
    6460 Mon Dec 28 20:02:22 2015
new/usr/src/man/man3c/priv_addset.3c
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
'\" te
.\" Copyright (c) 2005, Sun Microsystems, Inc. All Rights Reserved.
.\" The contents of this file are subject to the terms of the Common Development
.\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
.\" When distributing Covered Code, include this CDDL HEADER in each file and in
.TH PRIV_ADDSET 3C "Sep 08, 2015"
.SH NAME
priv_addset, priv_allocset, priv_copyset, priv_delset, priv_emptyset,
priv_basicset, priv_defaultset, priv_fillset, priv_freeset, priv_intersect,
priv_inverse, priv_isemptyset, priv_isequalset, priv_isfullset, priv_ismember,
priv_basicset, priv_fillset, priv_freeset, priv_intersect, priv_inverse,
priv_isemptyset, priv_isequalset, priv_isfullset, priv_ismember,
priv_issubset, priv_union \-
privilege set manipulation functions
.SH SYNOPSIS
.LP
.nf
#include <priv.h>

\fBint\fR \fBpriv_addset\fR(\fBpriv_set_t *\fR\fIsp\fR, \fBconst char *\fR\fIpri
.fi

.LP
.nf
\fBpriv_set_t *\fR\fBpriv_allocset\fR(\fBvoid\fR);
.fi

.LP
.nf
\fBvoid\fR \fBpriv_copyset\fR(\fBconst priv_set_t *\fR\fIsrc\fR, \fBpriv_set_t *
.fi

.LP
.nf
\fBint\fR \fBpriv_delset\fR(\fBpriv_set_t *\fR\fIsp\fR, \fBconst char *\fR\fIpri
.fi

.LP
.nf
\fBvoid\fR \fBpriv_emptyset\fR(\fBpriv_set_t *\fR\fIsp\fR);
.fi

.LP
.nf
\fBvoid\fR \fBpriv_basicset\fR(\fBpriv_set_t *\fR\fIsp\fR);
.fi

.LP
.nf
\fBvoid\fR \fBpriv_defaultset\fR(\fBpriv_set_t *\fR\fIsp\fR);
.fi

.LP
.nf
#endif /* ! codereview */
\fBvoid\fR \fBpriv_fillset\fR(\fBpriv_set_t *\fR\fIsp\fR);
.fi

.LP
.nf
\fBvoid\fR \fBpriv_freeset\fR(\fBpriv_set_t *\fR\fIsp\fR);
```



```
.fi

.LP
.nf
\fBvoid\fR \fBpriv_intersect\fR(\fBconst priv_set_t *\fR\fIsrc\fR, \fBpriv_set_t
.fi

.LP
.nf
\fBvoid\fR \fBpriv_inverse\fR(\fBpriv_set_t *\fR\fIsp\fR);
.fi

.LP
.nf
\fBboolean_t\fR \fBpriv_isemptyset\fR(\fBconst priv_set_t *\fR\fIsp\fR);
.fi

.LP
.nf
\fBboolean_t\fR \fBpriv_isequalset\fR(\fBconst priv_set_t *\fR\fIsrc\fR, \fBcons
.fi

.LP
.nf
\fBboolean_t\fR \fBpriv_isfullset\fR(\fBconst priv_set_t *\fR\fIsp\fR);
.fi

.LP
.nf
\fBboolean_t\fR \fBpriv_ismember\fR(\fBconst priv_set_t *\fR\fIsp\fR, \fBconst c
.fi

.LP
.nf
\fBboolean_t\fR \fBpriv_issubset\fR(\fBconst priv_set_t *\fR\fIsrc\fR, \fBconst
.fi

.LP
.nf
\fBvoid\fR \fBpriv_union\fR(\fBconst priv_set_t *\fR\fIsrc\fR, \fBpriv_set_t *\f
.fi

.SH DESCRIPTION
.LP
The \fIsp\fR, \fIsrc\fR, and \fIdst\fR arguments point to privilege sets. The
\fIpriv\fR argument points to a named privilege.
.sp
.LP
The \fBpriv_addset()\fR function adds the named privilege \fIpriv\fR to
\fIsp\fR.
.sp
.LP
The \fBpriv_allocset()\fR function allocates sufficient memory to contain a
privilege set. The value of the returned privilege set is indeterminate. The
function returns \fINULL\fR and sets \fBerrno\fR when it fails to allocate
memory.
.sp
.LP
The \fBpriv_copyset()\fR function copies the set \fIsrc\fR to \fIdst\fR.
.sp
.LP
The \fBpriv_delset()\fR function removes the named privilege \fIpriv\fR from
\fIsp\fR.
.sp
.LP
The \fBpriv_emptyset()\fR function clears all privileges from \fIsp\fR.
```

```
.sp
.LP
The \fBpriv_basicset()\fR function copies the basic privilege set to \fIsp\fR.
.sp
.LP
The \fBpriv_defaultset()\fR function copies the default privilege set to
\fIsp\fR.
.sp
.LP
#endif /* ! codereview */
The \fBpriv_fillset()\fR function asserts all privileges in \fIsp\fR, including
the privileges not currently defined in the system.
.sp
.LP
The \fBpriv_freeset()\fR function frees the storage allocated by
\fBpriv_allocset()\fR.
.sp
.LP
The \fBpriv_intersect()\fR function intersects \fIsrc\fR with \fIdst\fR and
places the results in \fIdst\fR.
.sp
.LP
The \fBpriv_inverse()\fR function inverts the privilege set given as argument
in place.
.sp
.LP
The \fBpriv_isemptyset()\fR function checks whether the argument is an empty
set.
.sp
.LP
The \fBpriv_isequalset()\fR function checks whether the privilege set \fIsrc\fR
is equal to \fIdst\fR.
.sp
.LP
The \fBpriv_isfullset()\fR function checks whether the argument is a full set.
A full set is a set with all bits set, regardless of whether the privilege is
currently defined in the system.
.sp
.LP
The \fBpriv_ismember()\fR function checks whether the named privilege
\fIpriv\fR is a member of \fIsp\fR.
.sp
.LP
The \fBpriv_issubset()\fR function checks whether \fIsrc\fR is a subset of
\fIdst\fR.
.sp
.LP
The \fBpriv_union()\fR function takes the union of \fIsrc\fR and \fIdst\fR and
places the result in \fIdst\fR.
.SH RETURN VALUES
.LP
Upon successful completion, \fBpriv_allocset()\fR returns a pointer to an
opaque data structure. It returns \fINULL\fR if memory allocation fails and
sets \fBerrno\fR to indicate the error.
.sp
.LP
Upon successful completion, \fBpriv_isemptyset()\fR, \fBpriv_isfullset()\fR,
\fBpriv_isequalset()\fR, \fBpriv_issubset()\fR, and \fBpriv_ismember()\fR
return \fBB_TRUE\fR. Otherwise, they return \fBB_FALSE\fR.
.sp
.LP
Upon successful completion, \fBpriv_delset()\fR and \fBpriv_addset()\fR return
0. Otherwise, they return -1 and set \fBerrno\fR to indicate the error.
.SH ERRORS
.LP
The \fBpriv_allocset()\fR function will fail if:
```



```
.sp
.ne 2
.na
\fB\fBENOMEM\fR\fR
.ad
.RS 10n
The physical limits of the system are exceeded by the memory allocation needed
to hold a privilege set.
.RE

.sp
.ne 2
.na
\fB\fBEAGAIN\fR\fR
.ad
.RS 10n
There is insufficient memory for allocation to hold a privilege set. The
application can try again later.
.RE

.sp
.LP
The \fBpriv_delset()\fR and \fBpriv_addset()\fR functions will fail if:
.sp
.ne 2
.na
\fB\fBEINVAL\fR\fR
.ad
.RS 10n
The privilege argument is not a valid privilege name.
.RE

.SH ATTRIBUTES
.LP
See \fBattributes\fR(5) for descriptions of the following attributes:
.sp

.sp
.TS
box;
c | c
l | l .
ATTRIBUTE TYPE  ATTRIBUTE VALUE
_
Interface Stability     Evolving
_
MT-Level        MT-Safe
.TE

.SH SEE ALSO
.LP
\fBsetppriv\fR(2), \fBmalloc\fR(3C), \fBpriv_str_to_set\fR(3C),
\fBattributes\fR(5), \fBprivileges\fR(5)
.SH NOTES
.LP
The functions that compare sets operate on all bits of the set, regardless of
whether the specific privileges are currently defined in the system.
```

```
**********************************************************
   34122 Mon Dec 28 20:02:30 2015
new/usr/src/man/man5/privileges.5
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
```

```
'\" te
.\" Copyright (c) 2009, Sun Microsystems, Inc. All Rights Reserved.
.\" Copyright 2015, Joyent, Inc. All Rights Reserved.
.\" The contents of this file are subject to the terms of the Common Development
.\"  See the License for the specific language governing permissions and limitat
.\" the fields enclosed by brackets "[]" replaced with your own identifying info
.TH PRIVILEGES 5 "Oct 30, 2015"
.SH NAME
privileges \- process privilege model
.SH DESCRIPTION
.LP
Solaris software implements a set of privileges that provide fine-grained
control over the actions of processes. The possession of a certain privilege
allows a process to perform a specific set of restricted operations.
.sp
.LP
The change to a primarily privilege-based security model in the Solaris
operating system gives developers an opportunity to restrict processes to those
privileged operations actually needed instead of all (super-user) or no
privileges (non-zero UIDs). Additionally, a set of previously unrestricted
operations now requires a privilege; these privileges are dubbed the "basic"
privileges.
privileges and are by default given to all processes.
.sp
.LP
The "basic" privileges, and certain privileges representing concepts not
traditionally present are, by default, given to all processes.  These are the
"default" set of privileges.
.sp
.LP
Taken together, all defined privileges with the exception of the "default"
Taken together, all defined privileges with the exception of the "basic"
privileges compose the set of privileges that are traditionally associated with
the root user. The "basic" privileges are "privileges" unprivileged processes
were accustomed to having, and the "default" privileges are the "basic"
privileges plus additions that while unprivileged processes aren't accustomed to
they should now have.
were accustomed to having.
.sp
.LP
The defined privileges are:
.sp
.ne 2
.na
\fB\fBPRIV_CONTRACT_EVENT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to request reliable delivery of events to an event endpoint.
.sp
Allow a process to include events in the critical event set term of a template
which could be generated in volume by the user.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_CONTRACT_IDENTITY\fR\fR
.ad
.sp .6
.RS 4n
```



```
Allows a process to set the service FMRI value of a process contract template.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_CONTRACT_OBSERVER\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to observe contract events generated by contracts created and
owned by users other than the process's effective user ID.
.sp
Allow a process to open contract event endpoints belonging to contracts created
and owned by users other than the process's effective user ID.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_CPC_CPU\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to access per-CPU hardware performance counters.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_DTRACE_KERNEL\fR\fR
.ad
.sp .6
.RS 4n
Allow DTrace kernel-level tracing.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_DTRACE_PROC\fR\fR
.ad
.sp .6
.RS 4n
Allow DTrace process-level tracing. Allow process-level tracing probes to be
placed and enabled in processes to which the user has permissions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_DTRACE_USER\fR\fR
.ad
.sp .6
.RS 4n
Allow DTrace user-level tracing. Allow use of the syscall and profile DTrace
providers to examine processes to which the user has permissions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_CHOWN\fR\fR
.ad
.sp .6
.RS 4n
```

```
125 Allow a process to change a file’s owner user ID. Allow a process to change a
126 file’s group ID to one other than the process’s effective group ID or one of
127 the process’s supplemental group IDs.
128 .RE

130 .sp
131 .ne 2
132 .na
133 \fB\fBPRIV_FILE_CHOWN_SELF\fR\fR
134 .ad
135 .sp .6
136 .RS 4n
137 Allow a process to give away its files. A process with this privilege runs as
138 if {\fB_POSIX_CHOWN_RESTRICTED\fR} is not in effect.
139 .RE

141 .sp
142 .ne 2
143 .na
144 \fB\fBPRIV_FILE_DAC_EXECUTE\fR\fR
145 .ad
146 .sp .6
147 .RS 4n
148 Allow a process to execute an executable file whose permission bits or ACL
149 would otherwise disallow the process execute permission.
150 .RE

152 .sp
153 .ne 2
154 .na
155 \fB\fBPRIV_FILE_DAC_READ\fR\fR
156 .ad
157 .sp .6
158 .RS 4n
159 Allow a process to read a file or directory whose permission bits or ACL would
160 otherwise disallow the process read permission.
161 .RE

163 .sp
164 .ne 2
165 .na
166 \fB\fBPRIV_FILE_DAC_SEARCH\fR\fR
167 .ad
168 .sp .6
169 .RS 4n
170 Allow a process to search a directory whose permission bits or ACL would not
171 otherwise allow the process search permission.
172 .RE

174 .sp
175 .ne 2
176 .na
177 \fB\fBPRIV_FILE_DAC_WRITE\fR\fR
178 .ad
179 .sp .6
180 .RS 4n
181 Allow a process to write a file or directory whose permission bits or ACL do
182 not allow the process write permission. All privileges are required to write
183 files owned by UID 0 in the absence of an effective UID of 0.
184 .RE

186 .sp
187 .ne 2
188 .na
189 \fB\fBPRIV_FILE_DOWNGRADE_SL\fR\fR
190 .ad
```



```
191 .sp .6
192 .RS 4n
193 Allow a process to set the sensitivity label of a file or directory to a
194 sensitivity label that does not dominate the existing sensitivity label.
195 .sp
196 This privilege is interpreted only if the system is configured with Trusted
197 Extensions.
198 .RE

200 .sp
201 .ne 2
202 .na
203 \fB\fBPRIV_FILE_FLAG_SET\fR\fR
204 .ad
205 .sp .6
206 .RS 4n
207 Allows a process to set immutable, nounlink or appendonly file attributes.
208 .RE

210 .sp
211 .ne 2
212 .na
213 \fB\fBPRIV_FILE_LINK_ANY\fR\fR
214 .ad
215 .sp .6
216 .RS 4n
217 Allow a process to create hardlinks to files owned by a UID different from the
218 process’s effective UID.
219 .RE

221 .sp
222 .ne 2
223 .na
224 \fB\fBPRIV_FILE_OWNER\fR\fR
225 .ad
226 .sp .6
227 .RS 4n
228 Allow a process that is not the owner of a file to modify that file’s access
229 and modification times. Allow a process that is not the owner of a directory to
230 modify that directory’s access and modification times. Allow a process that is
231 not the owner of a file or directory to remove or rename a file or directory
232 whose parent directory has the "save text image after execution" (sticky) bit
233 set. Allow a process that is not the owner of a file to mount a \fBnamefs\fR
234 upon that file. Allow a process that is not the owner of a file or directory to
235 modify that file’s or directory’s permission bits or ACL.
236 .RE

238 .sp
239 .ne 2
240 .na
241 \fB\fBPRIV_FILE_READ\fR\fR
242 .ad
243 .sp .6
244 .RS 4n
245 Allow a process to open objects in the filesystem for reading. This
246 privilege is not necessary to read from an already open file which was opened
247 before dropping the \fBPRIV_FILE_READ\fR privilege.
248 .RE

250 .sp
251 .ne 2
252 .na
253 \fB\fBPRIV_FILE_SETID\fR\fR
254 .ad
255 .sp .6
256 .RS 4n
```

```
257 Allow a process to change the ownership of a file or write to a file without
258 the set-user-ID and set-group-ID bits being cleared. Allow a process to set the
259 set-group-ID bit on a file or directory whose group is not the process's
260 effective group or one of the process's supplemental groups. Allow a process to
261 set the set-user-ID bit on a file with different ownership in the presence of
262 \fBPRIV_FILE_OWNER\fR. Additional restrictions apply when creating or modifying
263 a setuid 0 file.
264 .RE

266 .sp
267 .ne 2
268 .na
269 \fB\fBPRIV_FILE_UPGRADE_SL\fR\fR
270 .ad
271 .sp .6
272 .RS 4n
273 Allow a process to set the sensitivity label of a file or directory to a
274 sensitivity label that dominates the existing sensitivity label.
275 .sp
276 This privilege is interpreted only if the system is configured with Trusted
277 Extensions.
278 .RE

280 .sp
281 .ne 2
282 .na
283 \fB\fBPRIV_FILE_WRITE\fR\fR
284 .ad
285 .sp .6
286 .RS 4n
287 Allow a process to open objects in the filesytem for writing, or otherwise
288 modify them. This privilege is not necessary to write to an already open file
289 which was opened before dropping the \fBPRIV_FILE_WRITE\fR privilege.
290 .RE

292 .sp
293 .ne 2
294 .na
295 \fB\fBPRIV_GRAPHICS_ACCESS\fR\fR
296 .ad
297 .sp .6
298 .RS 4n
299 Allow a process to make privileged ioctls to graphics devices. Typically only
300 an xserver process needs to have this privilege. A process with this privilege
301 is also allowed to perform privileged graphics device mappings.
302 .RE

304 .sp
305 .ne 2
306 .na
307 \fB\fBPRIV_GRAPHICS_MAP\fR\fR
308 .ad
309 .sp .6
310 .RS 4n
311 Allow a process to perform privileged mappings through a graphics device.
312 .RE

314 .sp
315 .ne 2
316 .na
317 \fB\fBPRIV_IPC_DAC_READ\fR\fR
318 .ad
319 .sp .6
320 .RS 4n
321 Allow a process to read a System V IPC Message Queue, Semaphore Set, or Shared
322 Memory Segment whose permission bits would not otherwise allow the process read
```



```
323 permission.
324 .RE

326 .sp
327 .ne 2
328 .na
329 \fB\fBPRIV_IPC_DAC_WRITE\fR\fR
330 .ad
331 .sp .6
332 .RS 4n
333 Allow a process to write a System V IPC Message Queue, Semaphore Set, or Shared
334 Memory Segment whose permission bits would not otherwise allow the process
335 write permission.
336 .RE

338 .sp
339 .ne 2
340 .na
341 \fB\fBPRIV_IPC_OWNER\fR\fR
342 .ad
343 .sp .6
344 .RS 4n
345 Allow a process that is not the owner of a System V IPC Message Queue,
346 Semaphore Set, or Shared Memory Segment to remove, change ownership of, or
347 change permission bits of the Message Queue, Semaphore Set, or Shared Memory
348 Segment.
349 .RE

351 .sp
352 .ne 2
353 .na
354 \fB\fBPRIV_NET_ACCESS\fR\fR
355 .ad
356 .sp .6
357 .RS 4n
358 Allow a process to open a TCP, UDP, SDP, or SCTP network endpoint. This
359 privilege is not necessary to communicate using an existing endpoint already
360 opened before dropping the \fBPRIV_NET_ACCESS\fR privilege.
361 .RE

363 .sp
364 .ne 2
365 .na
366 \fB\fBPRIV_NET_BINDMLP\fR\fR
367 .ad
368 .sp .6
369 .RS 4n
370 Allow a process to bind to a port that is configured as a multi-level port
371 (MLP) for the process's zone. This privilege applies to both shared address and
372 zone-specific address MLPs. See \fBtnzonecfg\fR(\fB4\fR) from the Trusted
373 Extensions manual pages for information on configuring MLP ports.
374 .sp
375 This privilege is interpreted only if the system is configured with Trusted
376 Extensions.
377 .RE

379 .sp
380 .ne 2
381 .na
382 \fB\fBPRIV_NET_ICMPACCESS\fR\fR
383 .ad
384 .sp .6
385 .RS 4n
386 Allow a process to send and receive ICMP packets.
387 .RE
```

```
.sp
.ne 2
.na
\fB\fBPRIV_NET_MAC_AWARE\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to set the \fBNET_MAC_AWARE\fR process flag by using
\fBsetpflags\fR(2). This privilege also allows a process to set the
\fBSO_MAC_EXEMPT\fR socket option by using \fBsetsockopt\fR(3SOCKET). The
\fBNET_MAC_AWARE\fR process flag and the \fBSO_MAC_EXEMPT\fR socket option both
allow a local process to communicate with an unlabeled peer if the local
process's label dominates the peer's default label, or if the local process
runs in the global zone.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_NET_MAC_IMPLICIT\fR\fR
.ad
.sp .6
.RS 4n
Allow a proces to set \fBSO_MAC_IMPLICIT\fR option by using
\fBsetsockopt\fR(3SOCKET).  This allows a privileged process to transmit
implicitly-labeled packets to a peer.
.sp
This privilege is interpreted only if the system is configured with
Trusted Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_NET_OBSERVABILITY\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to open a device for just receiving network traffic, sending
traffic is disallowed.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_NET_PRIVADDR\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to bind to a privileged port number. The privilege port numbers
are 1-1023 (the traditional UNIX privileged ports) as well as those ports
marked as "\fBudp/tcp_extra_priv_ports\fR" with the exception of the ports
reserved for use by NFS and SMB.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_NET_RAWACCESS\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to have direct access to the network layer.
```



```
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_AUDIT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to generate audit records. Allow a process to get its own audit
pre-selection information.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_CHROOT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to change its root directory.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_CLOCK_HIGHRES\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to use high resolution timers.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_EXEC\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to call \fBexec\fR(2).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_FORK\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to call \fBfork\fR(2), \fBfork1\fR(2), or \fBvfork\fR(2).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_INFO\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to examine the status of processes other than those to which it
can send signals. Processes that cannot be examined cannot be seen in
\fB/proc\fR and appear not to exist.
.RE

.sp
```

```
.ne 2
.na
\fB\fBPRIV_PROC_LOCK_MEMORY\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to lock pages in physical memory.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_MEMINFO\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to access physical memory information.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_OWNER\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to send signals to other processes and inspect and modify the
process state in other processes, regardless of ownership. When modifying
another process, additional restrictions apply: the effective privilege set of
the attaching process must be a superset of the target process's effective,
permitted, and inheritable sets; the limit set must be a superset of the
target's limit set; if the target process has any UID set to 0 all privilege
must be asserted unless the effective UID is 0. Allow a process to bind
arbitrary processes to CPUs.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_PRIOUP\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to elevate its priority above its current level.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_PRIOCNTL\fR\fR
.ad
.sp .6
.RS 4n
Allows all that PRIV_PROC_PRIOUP allows.
Allow a process to change its scheduling class to any scheduling class,
including the RT class.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_SESSION\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to send signals or trace processes outside its session.
```



```
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_SETID\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to set its UIDs at will, assuming UID 0 requires all privileges
to be asserted.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_TASKID\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to assign a new task ID to the calling process.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_PROC_ZONE\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to trace or send signals to processes in other zones. See
\fBzones\fR(5).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_ACCT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to enable and disable and manage accounting through
\fBacct\fR(2).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_ADMIN\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to perform system administration tasks such as setting node and
domain name and specifying \fBcoreadm\fR(1M) and \fBnscd\fR(1M) settings
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_AUDIT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to start the (kernel) audit daemon. Allow a process to view and
set audit state (audit user ID, audit terminal ID, audit sessions ID, audit
pre-selection mask). Allow a process to turn off and on auditing. Allow a
```

process to configure the audit parameters (cache and queue sizes, event to class mappings, and policy options).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to perform various system configuration tasks. Allow filesystem-specific administrative procedures, such as filesystem configuration ioctls, quota calls, creation and deletion of snapshots, and manipulating the PCFS bootsector.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_DEVICES\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to create device special files. Allow a process to successfully call a kernel module that calls the kernel \fBdrv_priv\fR(9F) function to check for allowed access. Allow a process to open the real console device directly. Allow a process to open devices that have been exclusively opened.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_DL_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to configure a system’s datalink interfaces.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_IP_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to configure a system’s IP interfaces and routes. Allow a process to configure network parameters for \fBTCP/IP\fR using \fBndd\fR. Allow a process access to otherwise restricted \fBTCP/IP\fR information using \fBndd\fR. Allow a process to configure \fBIPsec\fR. Allow a process to pop anchored \fBSTREAM\fRs modules with matching \fBzoneid\fR.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_IPC_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to increase the size of a System V IPC Message Queue buffer.
.RE

.sp
.ne 2

.na
\fB\fBPRIV_SYS_IPTUN_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to configure IP tunnel links.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_LINKDIR\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to unlink and link directories.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_MOUNT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to mount and unmount filesystems that would otherwise be restricted (that is, most filesystems except \fBnamefs\fR). Allow a process to add and remove swap devices.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_NET_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to do all that \fBPRIV_SYS_IP_CONFIG\fR, \fBPRIV_SYS_DL_CONFIG\fR, and \fBPRIV_SYS_PPP_CONFIG\fR allow, plus the following: use the \fBrpcmod\fR STREAMS module and insert/remove STREAMS modules on locations other than the top of the module stack.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_NFS\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to provide NFS service: start NFS kernel threads, perform NFS locking operations, bind to NFS reserved ports: ports 2049 (\fBnfs\fR) and port 4045 (\fBlockd\fR).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_PPP_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to create, configure, and destroy PPP instances with pppd(1M) \fBpppd\fR(1M) and control PPPoE plumbing with \fBsppptun\fR(1M)sppptun(1M). This privilege is granted by default to exclusive IP stack instance zones.
.RE

```
.sp
.ne 2
.na
\fB\fBPRIV_SYS_RES_BIND\fR\fR
.ad
.sp .6
.RS 4n
Allows a process to bind processes to processor sets.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_RES_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allows all that PRIV_SYS_RES_BIND allows.
Allow a process to create and delete processor sets, assign CPUs to processor
sets and override the \fBPSET_NOESCAPE\fR property. Allow a process to change
the operational status of CPUs in the system using \fBp_online\fR(2). Allow a
process to configure filesystem quotas. Allow a process to configure resource
pools and bind processes to pools.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_RESOURCE\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to exceed the resource limits imposed on it by
\fBsetrlimit\fR(2) and \fBsetrctl\fR(2).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_SMB\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to provide NetBIOS or SMB services: start SMB kernel threads or
bind to NetBIOS or SMB reserved ports: ports 137, 138, 139 (NetBIOS) and 445
(SMB).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_SUSER_COMPAT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to successfully call a third party loadable module that calls
the kernel \fBsuser()\fR function to check for allowed access. This privilege
exists only for third party loadable module compatibility and is not used by
Solaris proper.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_TIME\fR\fR
```



```
.ad
.sp .6
.RS 4n
Allow a process to manipulate system time using any of the appropriate system
calls: \fBstime\fR(2), \fBadjtime\fR(2), and \fBntp_adjtime\fR(2).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_TRANS_LABEL\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to translate labels that are not dominated by the process's
sensitivity label to and from an external string form.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_VIRT_MANAGE\fR\fR
.ad
.sp .6
.RS 4n
Allows a process to manage virtualized environments such as \fBxVM\fR(5).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_COLORMAP\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to override colormap restrictions.
.sp
Allow a process to install or remove colormaps.
.sp
Allow a process to retrieve colormap cell entries allocated by other processes.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to configure or destroy resources that are permanently retained
by the X server.
.sp
Allow a process to use SetScreenSaver to set the screen saver timeout value
.sp
Allow a process to use ChangeHosts to modify the display access control list.
.sp
Allow a process to use GrabServer.
.sp
Allow a process to use the SetCloseDownMode request that can retain window,
pixmap, colormap, property, cursor, font, or graphic context resources.
```

```
 917 .sp
 918 This privilege is interpreted only if the system is configured with Trusted
 919 Extensions.
 920 .RE

 922 .sp
 923 .ne 2
 924 .na
 925 \fB\fBPRIV_WIN_DAC_READ\fR\fR
 926 .ad
 927 .sp .6
 928 .RS 4n
 929 Allow a process to read from a window resource that it does not own (has a
 930 different user ID).
 931 .sp
 932 This privilege is interpreted only if the system is configured with Trusted
 933 Extensions.
 934 .RE

 936 .sp
 937 .ne 2
 938 .na
 939 \fB\fBPRIV_WIN_DAC_WRITE\fR\fR
 940 .ad
 941 .sp .6
 942 .RS 4n
 943 Allow a process to write to or create a window resource that it does not own
 944 (has a different user ID). A newly created window property is created with the
 945 window's user ID.
 946 .sp
 947 This privilege is interpreted only if the system is configured with Trusted
 948 Extensions.
 949 .RE

 951 .sp
 952 .ne 2
 953 .na
 954 \fB\fBPRIV_WIN_DEVICES\fR\fR
 955 .ad
 956 .sp .6
 957 .RS 4n
 958 Allow a process to perform operations on window input devices.
 959 .sp
 960 Allow a process to get and set keyboard and pointer controls.
 961 .sp
 962 Allow a process to modify pointer button and key mappings.
 963 .sp
 964 This privilege is interpreted only if the system is configured with Trusted
 965 Extensions.
 966 .RE

 968 .sp
 969 .ne 2
 970 .na
 971 \fB\fBPRIV_WIN_DGA\fR\fR
 972 .ad
 973 .sp .6
 974 .RS 4n
 975 Allow a process to use the direct graphics access (DGA) X protocol extensions.
 976 Direct process access to the frame buffer is still required. Thus the process
 977 must have MAC and DAC privileges that allow access to the frame buffer, or the
 978 frame buffer must be allocated to the process.
 979 .sp
 980 This privilege is interpreted only if the system is configured with Trusted
 981 Extensions.
 982 .RE
```



```
 984 .sp
 985 .ne 2
 986 .na
 987 \fB\fBPRIV_WIN_DOWNGRADE_SL\fR\fR
 988 .ad
 989 .sp .6
 990 .RS 4n
 991 Allow a process to set the sensitivity label of a window resource to a
 992 sensitivity label that does not dominate the existing sensitivity label.
 993 .sp
 994 This privilege is interpreted only if the system is configured with Trusted
 995 Extensions.
 996 .RE

 998 .sp
 999 .ne 2
1000 .na
1001 \fB\fBPRIV_WIN_FONTPATH\fR\fR
1002 .ad
1003 .sp .6
1004 .RS 4n
1005 Allow a process to set a font path.
1006 .sp
1007 This privilege is interpreted only if the system is configured with Trusted
1008 Extensions.
1009 .RE

1011 .sp
1012 .ne 2
1013 .na
1014 \fB\fBPRIV_WIN_MAC_READ\fR\fR
1015 .ad
1016 .sp .6
1017 .RS 4n
1018 Allow a process to read from a window resource whose sensitivity label is not
1019 equal to the process sensitivity label.
1020 .sp
1021 This privilege is interpreted only if the system is configured with Trusted
1022 Extensions.
1023 .RE

1025 .sp
1026 .ne 2
1027 .na
1028 \fB\fBPRIV_WIN_MAC_WRITE\fR\fR
1029 .ad
1030 .sp .6
1031 .RS 4n
1032 Allow a process to create a window resource whose sensitivity label is not
1033 equal to the process sensitivity label. A newly created window property is
1034 created with the window's sensitivity label.
1035 .sp
1036 This privilege is interpreted only if the system is configured with Trusted
1037 Extensions.
1038 .RE

1040 .sp
1041 .ne 2
1042 .na
1043 \fB\fBPRIV_WIN_SELECTION\fR\fR
1044 .ad
1045 .sp .6
1046 .RS 4n
1047 Allow a process to request inter-window data moves without the intervention of
1048 the selection confirmer.
```

```
1049 .sp
1050 This privilege is interpreted only if the system is configured with Trusted
1051 Extensions.
1052 .RE

1054 .sp
1055 .ne 2
1056 .na
1057 \fB\fBPRIV_WIN_UPGRADE_SL\fR\fR
1058 .ad
1059 .sp .6
1060 .RS 4n
1061 Allow a process to set the sensitivity label of a window resource to a
1062 sensitivity label that dominates the existing sensitivity label.
1063 .sp
1064 This privilege is interpreted only if the system is configured with Trusted
1065 Extensions.
1066 .RE

1068 .sp
1069 .ne 2
1070 .na
1071 \fB\fBPRIV_XVM_CONTROL\fR\fR
1072 .ad
1073 .sp .6
1074 .RS 4n
1075 Allows a process access to the \fBxVM\fR(5) control devices for managing guest
1076 domains and the hypervisor. This privilege is used only if booted into xVM on
1077 x86 platforms.
1078 .RE

1080 .sp
1081 .LP
1082 Of the privileges listed above, the privileges \fBPRIV_FILE_LINK_ANY\fR,
1083 \fBPRIV_PROC_INFO\fR, \fBPRIV_PROC_SESSION\fR, \fBPRIV_PROC_FORK\fR,
1084 \fBPRIV_FILE_READ\fR, \fBPRIV_FILE_WRITE\fR, \fBPRIV_NET_ACCESS\fR and
1085 \fBPRIV_PROC_EXEC\fR are considered "basic" privileges. These are privileges
1086 that used to be always available to unprivileged processes. By default,
1087 processes still have the basic privileges.
1088 .sp
1089 .LP
1090 The privileges \fBPRIV_PROC_SETID\fR and \fBPRIV_PROC_AUDIT\fR must be present
1091 in the Limit set (see below) of a process in order for set-uid root \fBexec\fRs
1092 to be successful, that is, get an effective UID of 0 and additional privileges.
1093 .sp
1094 .LP
1095 The privilege implementation in Solaris extends the process credential with
1096 four privilege sets:
1097 .sp
1098 .ne 2
1099 .na
1100 \fBI, the inheritable set\fR
1101 .ad
1102 .RS 26n
1103 The privileges inherited on \fBexec\fR.
1104 .RE

1106 .sp
1107 .ne 2
1108 .na
1109 \fBP, the permitted set\fR
1110 .ad
1111 .RS 26n
1112 The maximum set of privileges for the process.
1113 .RE
```



```
1115 .sp
1116 .ne 2
1117 .na
1118 \fBE, the effective set\fR
1119 .ad
1120 .RS 26n
1121 The privileges currently in effect.
1122 .RE

1124 .sp
1125 .ne 2
1126 .na
1127 \fBL, the limit set\fR
1128 .ad
1129 .RS 26n
1130 The upper bound of the privileges a process and its offspring can obtain.
1131 Changes to L take effect on the next \fBexec\fR.
1132 .RE

1134 .sp
1135 .LP
1136 The sets I, P and E are typically identical to the basic set of privileges for
1137 unprivileged processes. The limit set is typically the full set of privileges.
1138 .sp
1139 .LP
1140 Each process has a Privilege Awareness State (PAS) that can take the value PA
1141 (privilege-aware) and NPA (not-PA). PAS is a transitional mechanism that allows
1142 a choice between full compatibility with the old superuser model and completely
1143 ignoring the effective UID.
1144 .sp
1145 .LP
1146 To facilitate the discussion, we introduce the notion of "observed effective
1147 set" (oE) and "observed permitted set" (oP) and the implementation sets iE and
1148 iP.
1149 .sp
1150 .LP
1151 A process becomes privilege-aware either by manipulating the effective,
1152 permitted, or limit privilege sets through \fBsetppriv\fR(2) or by using
1153 \fBsetpflags\fR(2). In all cases, oE and oP are invariant in the process of
1154 becoming privilege-aware. In the process of becoming privilege-aware, the
1155 following assignments take place:
1156 .sp
1157 .in +2
1158 .nf
1159 iE = oE
1160 iP = oP
1161 .fi
1162 .in -2

1164 .sp
1165 .LP
1166 When a process is privilege-aware, oE and oP are invariant under UID changes.
1167 When a process is not privilege-aware, oE and oP are observed as follows:
1168 .sp
1169 .in +2
1170 .nf
1171 oE = euid == 0 ? L : iE
1172 oP = (euid == 0 || ruid == 0 || suid == 0) ? L : iP
1173 .fi
1174 .in -2

1176 .sp
1177 .LP
1178 When a non-privilege-aware process has an effective UID of 0, it can exercise
1179 the privileges contained in its limit set, the upper bound of its privileges.
1180 If a non-privilege-aware process has any of the UIDs 0, it appears to be
```

capable of potentially exercising all privileges in L.
.sp
.LP
It is possible for a process to return to the non-privilege aware state using \fBsetpflags()\fR. The kernel always attempts this on \fBexec\fR(2). This operation is permitted only if the following conditions are met:
.RS +4
.TP
.ie t \(bu
.el o
If any of the UIDs is equal to 0, P must be equal to L.
.RE
.RS +4
.TP
.ie t \(bu
.el o
If the effective UID is equal to 0, E must be equal to L.
.RE
.sp
.LP
When a process gives up privilege awareness, the following assignments take place:
.sp
.in +2
.nf
if (euid == 0) iE = L & I
if (any uid == 0) iP = L & I
.fi
.in -2

.sp
.LP
The privileges obtained when not having a UID of \fB0\fR are the inheritable set of the process restricted by the limit set.
.sp
.LP
Only privileges in the process's (observed) effective privilege set allow the process to perform restricted operations. A process can use any of the privilege manipulation functions to add or remove privileges from the privilege sets. Privileges can be removed always. Only privileges found in the permitted set can be added to the effective and inheritable set. The limit set cannot grow. The inheritable set can be larger than the permitted set.
.sp
.LP
When a process performs an \fBexec\fR(2), the kernel first tries to relinquish privilege awareness before making the following privilege set modifications:
.sp
.in +2
.nf
E' = P' = I' = L & I
L is unchanged
.fi
.in -2

.sp
.LP
If a process has not manipulated its privileges, the privilege sets effectively remain the same, as E, P and I are already identical.
.sp
.LP
The limit set is enforced at \fBexec\fR time.
.sp
.LP
To run a non-privilege-aware application in a backward-compatible manner, a privilege-aware application should start the non-privilege-aware application with I=basic.

.sp
.LP
For most privileges, absence of the privilege simply results in a failure. In some instances, the absense of a privilege can cause system calls to behave differently. In other instances, the removal of a privilege can force a set-uid application to seriously malfunction. Privileges of this type are considered "unsafe". When a process is lacking any of the unsafe privileges from its limit set, the system does not honor the set-uid bit of set-uid root applications. The following unsafe privileges have been identified: \fBproc_setid\fR, \fBsys_resource\fR and \fBproc_audit\fR.
.SS "Privilege Escalation"
.LP
In certain circumstances, a single privilege could lead to a process gaining one or more additional privileges that were not explicitly granted to that process. To prevent such an escalation of privileges, the security policy requires explicit permission for those additional privileges.
.sp
.LP
Common examples of escalation are those mechanisms that allow modification of system resources through "raw'' interfaces; for example, changing kernel data structures through \fB/dev/kmem\fR or changing files through \fB/dev/dsk/*\fR. Escalation also occurs when a process controls processes with more privileges than the controlling process. A special case of this is manipulating or creating objects owned by UID 0 or trying to obtain UID 0 using \fBsetuid\fR(2). The special treatment of UID 0 is needed because the UID 0 owns all system configuration files and ordinary file protection mechanisms allow processes with UID 0 to modify the system configuration. With appropriate file modifications, a given process running with an effective UID of 0 can gain all privileges.
.sp
.LP
In situations where a process might obtain UID 0, the security policy requires additional privileges, up to the full set of privileges. Such restrictions could be relaxed or removed at such time as additional mechanisms for protection of system files became available. There are no such mechanisms in the current Solaris release.
.sp
.LP
The use of UID 0 processes should be limited as much as possible. They should be replaced with programs running under a different UID but with exactly the privileges they need.
.sp
.LP
Daemons that never need to \fBexec\fR subprocesses should remove the \fBPRIV_PROC_EXEC\fR privilege from their permitted and limit sets.
.SS "Assigned Privileges and Safeguards"
.LP
When privileges are assigned to a user, the system administrator could give that user more powers than intended. The administrator should consider whether safeguards are needed. For example, if the \fBPRIV_PROC_LOCK_MEMORY\fR privilege is given to a user, the administrator should consider setting the \fBproject.max-locked-memory\fR resource control as well, to prevent that user from locking all memory.
.SS "Privilege Debugging"
.LP
When a system call fails with a permission error, it is not always immediately obvious what caused the problem. To debug such a problem, you can use a tool called \fBprivilege debugging\fR. When privilege debugging is enabled for a process, the kernel reports missing privileges on the controlling terminal of the process. (Enable debugging for a process with the \fB-D\fR option of \fBppriv\fR(1).) Additionally, the administrator can enable system-wide privilege debugging by setting the \fBsystem\fR(4) variable \fBpriv_debug\fR using:
.sp
.in +2
.nf

```
1313 set priv_debug = 1
1314 .fi
1315 .in -2

1317 .sp
1318 .LP
1319 On a running system, you can use \fBmdb\fR(1) to change this variable.
1320 .SS "Privilege Administration"
1321 .LP
1322 The Solaris Management Console (see \fBsmc\fR(1M)) is the preferred method of
1323 modifying privileges for a command. Use \fBusermod\fR(1M) or \fBsmrole\fR(1M)
1324 to assign privileges to or modify privileges for, respectively, a user or a
1325 role. Use \fBppriv\fR(1) to enumerate the privileges supported on a system and
1326 \fBtruss\fR(1) to determine which privileges a program requires.
1327 .SH SEE ALSO
1328 .LP
1329 \fBmdb\fR(1), \fBppriv\fR(1), \fBadd_drv\fR(1M), \fBifconfig\fR(1M),
1330 \fBlockd\fR(1M), \fBnfsd\fR(1M), \fBpppd\fR(1M), \fBrem_drv\fR(1M),
1331 \fBsmbd\fR(1M), \fBsppptun\fR(1M), \fBupdate_drv\fR(1M), \fBIntro\fR(2),
1332 \fBaccess\fR(2), \fBacct\fR(2), \fBacl\fR(2), \fBadjtime\fR(2), \fBaudit\fR(2),
1333 \fBauditon\fR(2), \fBchmod\fR(2), \fBchown\fR(2), \fBchroot\fR(2),
1334 \fBcreat\fR(2), \fBexec\fR(2), \fBfcntl\fR(2), \fBfork\fR(2),
1335 \fBfpathconf\fR(2), \fBgetacct\fR(2), \fBgetpflags\fR(2), \fBgetppriv\fR(2),
1336 \fBgetsid\fR(2), \fBkill\fR(2), \fBlink\fR(2), \fBmemcntl\fR(2),
1337 \fBmknod\fR(2), \fBmount\fR(2), \fBmsgctl\fR(2), \fBnice\fR(2),
1338 \fBntp_adjtime\fR(2), \fBopen\fR(2), \fBp_online\fR(2), \fBpriocntl\fR(2),
1339 \fBpriocntlset\fR(2), \fBprocessor_bind\fR(2), \fBpset_bind\fR(2),
1340 \fBpset_create\fR(2), \fBreadlink\fR(2), \fBresolvepath\fR(2), \fBrmdir\fR(2),
1341 \fBsemctl\fR(2), \fBsetauid\fR(2), \fBsetegid\fR(2), \fBseteuid\fR(2),
1342 \fBsetgid\fR(2), \fBsetgroups\fR(2), \fBsetpflags\fR(2), \fBsetppriv\fR(2),
1343 \fBsetrctl\fR(2), \fBsetregid\fR(2), \fBsetreuid\fR(2), \fBsetrlimit\fR(2),
1344 \fBsettaskid\fR(2), \fBsetuid\fR(2), \fBshmctl\fR(2), \fBshmget\fR(2),
1345 \fBshmop\fR(2), \fBsigsend\fR(2), \fBstat\fR(2), \fBstatvfs\fR(2),
1346 \fBstime\fR(2), \fBswapctl\fR(2), \fBsysinfo\fR(2), \fBuadmin\fR(2),
1347 \fBulimit\fR(2), \fBumount\fR(2), \fBunlink\fR(2), \fButime\fR(2),
1348 \fButimes\fR(2), \fBbind\fR(3SOCKET), \fBdoor_ucred\fR(3C),
1349 \fBpriv_addset\fR(3C), \fBpriv_set\fR(3C), \fBpriv_getbyname\fR(3C),
1350 \fBpriv_getbynum\fR(3C), \fBpriv_set_to_str\fR(3C), \fBpriv_str_to_set\fR(3C),
1351 \fBsocket\fR(3SOCKET), \fBt_bind\fR(3NSL), \fBtimer_create\fR(3C),
1352 \fBucred_get\fR(3C), \fBexec_attr\fR(4), \fBproc\fR(4), \fBsystem\fR(4),
1353 \fBuser_attr\fR(4), \fBxVM\fR(5), \fBddi_cred\fR(9F), \fBdrv_priv\fR(9F),
1354 \fBpriv_getbyname\fR(9F), \fBpriv_policy\fR(9F), \fBpriv_policy_choice\fR(9F),
1355 \fBpriv_policy_only\fR(9F)
1356 .sp
1357 .LP
1358 \fISystem Administration Guide: Security Services\fR
```

```
**********************************************************
   76690 Mon Dec 28 20:02:37 2015
new/usr/src/pkg/manifests/system-library.man3c.inc
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
```

```
   1 #
   2 # This file and its contents are supplied under the terms of the
   3 # Common Development and Distribution License ("CDDL"), version 1.0.
   4 # You may only use this file in accordance with the terms of version
   5 # 1.0 of the CDDL.
   6 #
   7 # A full copy of the text of the CDDL should have accompanied this
   8 # source.  A copy of the CDDL is also available via the Internet
   9 # at http://www.illumos.org/license/CDDL.
  10 #

  12 #
  13 # Copyright 2011, Richard Lowe
  14 # Copyright 2013 Nexenta Systems, Inc.  All rights reserved.
  15 # Copyright 2013 OmniTI Computer Consulting, Inc. All rights reserved.
  16 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
  17 #

  19 file path=usr/share/man/man3c/__fbufsize.3c
  20 file path=usr/share/man/man3c/_longjmp.3c
  21 file path=usr/share/man/man3c/_stack_grow.3c
  22 file path=usr/share/man/man3c/a64l.3c
  23 file path=usr/share/man/man3c/abort.3c
  24 file path=usr/share/man/man3c/abs.3c
  25 file path=usr/share/man/man3c/addsev.3c
  26 file path=usr/share/man/man3c/addseverity.3c
  27 file path=usr/share/man/man3c/aio_cancel.3c
  28 file path=usr/share/man/man3c/aio_error.3c
  29 file path=usr/share/man/man3c/aio_fsync.3c
  30 file path=usr/share/man/man3c/aio_read.3c
  31 file path=usr/share/man/man3c/aio_return.3c
  32 file path=usr/share/man/man3c/aio_suspend.3c
  33 file path=usr/share/man/man3c/aio_waitn.3c
  34 file path=usr/share/man/man3c/aio_write.3c
  35 file path=usr/share/man/man3c/aiocancel.3c
  36 file path=usr/share/man/man3c/aioread.3c
  37 file path=usr/share/man/man3c/aiowait.3c
  38 file path=usr/share/man/man3c/assert.3c
  39 file path=usr/share/man/man3c/arc4random.3c
  40 file path=usr/share/man/man3c/atexit.3c
  41 file path=usr/share/man/man3c/atomic_add.3c
  42 file path=usr/share/man/man3c/atomic_and.3c
  43 file path=usr/share/man/man3c/atomic_bits.3c
  44 file path=usr/share/man/man3c/atomic_cas.3c
  45 file path=usr/share/man/man3c/atomic_dec.3c
  46 file path=usr/share/man/man3c/atomic_inc.3c
  47 file path=usr/share/man/man3c/atomic_ops.3c
  48 file path=usr/share/man/man3c/atomic_or.3c
  49 file path=usr/share/man/man3c/atomic_swap.3c
  50 file path=usr/share/man/man3c/attropen.3c
  51 file path=usr/share/man/man3c/basename.3c
  52 file path=usr/share/man/man3c/bsd_signal.3c
  53 file path=usr/share/man/man3c/bsearch.3c
  54 file path=usr/share/man/man3c/bstring.3c
  55 file path=usr/share/man/man3c/btowc.3c
  56 file path=usr/share/man/man3c/catgets.3c
  57 file path=usr/share/man/man3c/catopen.3c
  58 file path=usr/share/man/man3c/cfgetispeed.3c
  59 file path=usr/share/man/man3c/cfsetispeed.3c
  60 file path=usr/share/man/man3c/clearenv.3c
  61 file path=usr/share/man/man3c/clock.3c
```



```
  62 file path=usr/share/man/man3c/clock_nanosleep.3c
  63 file path=usr/share/man/man3c/clock_settime.3c
  64 file path=usr/share/man/man3c/closedir.3c
  65 file path=usr/share/man/man3c/closefrom.3c
  66 file path=usr/share/man/man3c/cond_init.3c
  67 file path=usr/share/man/man3c/confstr.3c
  68 file path=usr/share/man/man3c/crypt.3c
  69 file path=usr/share/man/man3c/crypt_genhash_impl.3c
  70 file path=usr/share/man/man3c/crypt_gensalt.3c
  71 file path=usr/share/man/man3c/crypt_gensalt_impl.3c
  72 file path=usr/share/man/man3c/cset.3c
  73 file path=usr/share/man/man3c/ctermid.3c
  74 file path=usr/share/man/man3c/ctime.3c
  75 file path=usr/share/man/man3c/ctype.3c
  76 file path=usr/share/man/man3c/cuserid.3c
  77 file path=usr/share/man/man3c/daemon.3c
  78 file path=usr/share/man/man3c/decimal_to_floating.3c
  79 file path=usr/share/man/man3c/difftime.3c
  80 file path=usr/share/man/man3c/directio.3c
  81 file path=usr/share/man/man3c/dirfd.3c
  82 file path=usr/share/man/man3c/dirname.3c
  83 file path=usr/share/man/man3c/div.3c
  84 file path=usr/share/man/man3c/dladdr.3c
  85 file path=usr/share/man/man3c/dlclose.3c
  86 file path=usr/share/man/man3c/dldump.3c
  87 file path=usr/share/man/man3c/dlerror.3c
  88 file path=usr/share/man/man3c/dlinfo.3c
  89 file path=usr/share/man/man3c/dlopen.3c
  90 file path=usr/share/man/man3c/dlsym.3c
  91 file path=usr/share/man/man3c/door_bind.3c
  92 file path=usr/share/man/man3c/door_call.3c
  93 file path=usr/share/man/man3c/door_create.3c
  94 file path=usr/share/man/man3c/door_cred.3c
  95 file path=usr/share/man/man3c/door_getparam.3c
  96 file path=usr/share/man/man3c/door_info.3c
  97 file path=usr/share/man/man3c/door_return.3c
  98 file path=usr/share/man/man3c/door_revoke.3c
  99 file path=usr/share/man/man3c/door_server_create.3c
 100 file path=usr/share/man/man3c/door_ucred.3c
 101 file path=usr/share/man/man3c/drand48.3c
 102 file path=usr/share/man/man3c/dup2.3c
 103 file path=usr/share/man/man3c/econvert.3c
 104 file path=usr/share/man/man3c/ecvt.3c
 105 file path=usr/share/man/man3c/enable_extended_FILE_stdio.3c
 106 file path=usr/share/man/man3c/encrypt.3c
 107 file path=usr/share/man/man3c/end.3c
 108 file path=usr/share/man/man3c/epoll_create.3c
 109 file path=usr/share/man/man3c/epoll_ctl.3c
 110 file path=usr/share/man/man3c/epoll_wait.3c
 111 file path=usr/share/man/man3c/err.3c
 112 file path=usr/share/man/man3c/euclen.3c
 113 file path=usr/share/man/man3c/eventfd.3c
 114 file path=usr/share/man/man3c/exit.3c
 115 file path=usr/share/man/man3c/fattach.3c
 116 file path=usr/share/man/man3c/fclose.3c
 117 file path=usr/share/man/man3c/fcloseall.3c
 118 file path=usr/share/man/man3c/fdatasync.3c
 119 file path=usr/share/man/man3c/fdetach.3c
 120 file path=usr/share/man/man3c/fdopen.3c
 121 file path=usr/share/man/man3c/ferror.3c
 122 file path=usr/share/man/man3c/fflush.3c
 123 file path=usr/share/man/man3c/ffs.3c
 124 file path=usr/share/man/man3c/fgetattr.3c
 125 file path=usr/share/man/man3c/fgetc.3c
 126 file path=usr/share/man/man3c/fgetpos.3c
 127 file path=usr/share/man/man3c/fgetwc.3c
```

```
128 file path=usr/share/man/man3c/floating_to_decimal.3c
129 file path=usr/share/man/man3c/flock.3c
130 file path=usr/share/man/man3c/flockfile.3c
131 file path=usr/share/man/man3c/fmtmsg.3c
132 file path=usr/share/man/man3c/fnmatch.3c
133 file path=usr/share/man/man3c/fopen.3c
134 file path=usr/share/man/man3c/fpgetround.3c
135 file path=usr/share/man/man3c/fputc.3c
136 file path=usr/share/man/man3c/fputwc.3c
137 file path=usr/share/man/man3c/fputws.3c
138 file path=usr/share/man/man3c/fread.3c
139 file path=usr/share/man/man3c/freopen.3c
140 file path=usr/share/man/man3c/fseek.3c
141 file path=usr/share/man/man3c/fsetpos.3c
142 file path=usr/share/man/man3c/fsync.3c
143 file path=usr/share/man/man3c/ftell.3c
144 file path=usr/share/man/man3c/ftime.3c
145 file path=usr/share/man/man3c/ftok.3c
146 file path=usr/share/man/man3c/ftw.3c
147 file path=usr/share/man/man3c/fwide.3c
148 file path=usr/share/man/man3c/fwprintf.3c
149 file path=usr/share/man/man3c/fwrite.3c
150 file path=usr/share/man/man3c/fwscanf.3c
151 file path=usr/share/man/man3c/getcpuid.3c
152 file path=usr/share/man/man3c/getcwd.3c
153 file path=usr/share/man/man3c/getdate.3c
154 file path=usr/share/man/man3c/getdtablesize.3c
155 file path=usr/share/man/man3c/getentropy.3c
156 file path=usr/share/man/man3c/getenv.3c
157 file path=usr/share/man/man3c/getexecname.3c
158 file path=usr/share/man/man3c/getgrnam.3c
159 file path=usr/share/man/man3c/gethostid.3c
160 file path=usr/share/man/man3c/gethostname.3c
161 file path=usr/share/man/man3c/gethrtime.3c
162 file path=usr/share/man/man3c/getline.3c
163 file path=usr/share/man/man3c/getloadavg.3c
164 file path=usr/share/man/man3c/getlogin.3c
165 file path=usr/share/man/man3c/getmntent.3c
166 file path=usr/share/man/man3c/getnetgrent.3c
167 file path=usr/share/man/man3c/get_nprocs.3c
168 file path=usr/share/man/man3c/getopt.3c
169 file path=usr/share/man/man3c/getpagesize.3c
170 file path=usr/share/man/man3c/getpagesizes.3c
171 file path=usr/share/man/man3c/getpass.3c
172 file path=usr/share/man/man3c/getpeerucred.3c
173 file path=usr/share/man/man3c/getpriority.3c
174 file path=usr/share/man/man3c/getprogname.3c
175 file path=usr/share/man/man3c/getpw.3c
176 file path=usr/share/man/man3c/getpwnam.3c
177 file path=usr/share/man/man3c/getrusage.3c
178 file path=usr/share/man/man3c/gets.3c
179 file path=usr/share/man/man3c/getspnam.3c
180 file path=usr/share/man/man3c/getsubopt.3c
181 file path=usr/share/man/man3c/gettext.3c
182 file path=usr/share/man/man3c/gettimeofday.3c
183 file path=usr/share/man/man3c/gettxt.3c
184 file path=usr/share/man/man3c/getusershell.3c
185 file path=usr/share/man/man3c/getutent.3c
186 file path=usr/share/man/man3c/getutxent.3c
187 file path=usr/share/man/man3c/getvfsent.3c
188 file path=usr/share/man/man3c/getwc.3c
189 file path=usr/share/man/man3c/getwchar.3c
190 file path=usr/share/man/man3c/getwd.3c
191 file path=usr/share/man/man3c/getwidth.3c
192 file path=usr/share/man/man3c/getws.3c
193 file path=usr/share/man/man3c/getzoneid.3c
```



```
194 file path=usr/share/man/man3c/glob.3c
195 file path=usr/share/man/man3c/grantpt.3c
196 file path=usr/share/man/man3c/hsearch.3c
197 file path=usr/share/man/man3c/iconv.3c
198 file path=usr/share/man/man3c/iconv_close.3c
199 file path=usr/share/man/man3c/iconv_open.3c
200 file path=usr/share/man/man3c/imaxabs.3c
201 file path=usr/share/man/man3c/imaxdiv.3c
202 file path=usr/share/man/man3c/index.3c
203 file path=usr/share/man/man3c/initgroups.3c
204 file path=usr/share/man/man3c/insque.3c
205 file path=usr/share/man/man3c/is_system_labeled.3c
206 file path=usr/share/man/man3c/isaexec.3c
207 file path=usr/share/man/man3c/isastream.3c
208 file path=usr/share/man/man3c/isatty.3c
209 file path=usr/share/man/man3c/isnand.3c
210 file path=usr/share/man/man3c/iswalpha.3c
211 file path=usr/share/man/man3c/iswctype.3c
212 file path=usr/share/man/man3c/killpg.3c
213 file path=usr/share/man/man3c/lckpwdf.3c
214 file path=usr/share/man/man3c/lfmt.3c
215 file path=usr/share/man/man3c/lio_listio.3c
216 file path=usr/share/man/man3c/localeconv.3c
217 file path=usr/share/man/man3c/lockf.3c
218 file path=usr/share/man/man3c/lsearch.3c
219 file path=usr/share/man/man3c/madvise.3c
220 file path=usr/share/man/man3c/makecontext.3c
221 file path=usr/share/man/man3c/makedev.3c
222 file path=usr/share/man/man3c/malloc.3c
223 file path=usr/share/man/man3c/mblen.3c
224 file path=usr/share/man/man3c/mbrlen.3c
225 file path=usr/share/man/man3c/mbrtowc.3c
226 file path=usr/share/man/man3c/mbsinit.3c
227 file path=usr/share/man/man3c/mbsrtowcs.3c
228 file path=usr/share/man/man3c/mbtowc.3c
229 file path=usr/share/man/man3c/membar_ops.3c
230 file path=usr/share/man/man3c/memory.3c
231 file path=usr/share/man/man3c/mkfifo.3c
232 file path=usr/share/man/man3c/mkstemp.3c
233 file path=usr/share/man/man3c/mktemp.3c
234 file path=usr/share/man/man3c/mktime.3c
235 file path=usr/share/man/man3c/mlock.3c
236 file path=usr/share/man/man3c/mlockall.3c
237 file path=usr/share/man/man3c/monitor.3c
238 file path=usr/share/man/man3c/mq_close.3c
239 file path=usr/share/man/man3c/mq_getattr.3c
240 file path=usr/share/man/man3c/mq_notify.3c
241 file path=usr/share/man/man3c/mq_open.3c
242 file path=usr/share/man/man3c/mq_receive.3c
243 file path=usr/share/man/man3c/mq_send.3c
244 file path=usr/share/man/man3c/mq_setattr.3c
245 file path=usr/share/man/man3c/mq_unlink.3c
246 file path=usr/share/man/man3c/msync.3c
247 file path=usr/share/man/man3c/mutex_init.3c
248 file path=usr/share/man/man3c/nanosleep.3c
249 file path=usr/share/man/man3c/ndbm.3c
250 file path=usr/share/man/man3c/newlocale.3c
251 file path=usr/share/man/man3c/nl_langinfo.3c
252 file path=usr/share/man/man3c/offsetof.3c
253 file path=usr/share/man/man3c/opendir.3c
254 file path=usr/share/man/man3c/perror.3c
255 file path=usr/share/man/man3c/pfmt.3c
256 file path=usr/share/man/man3c/plock.3c
257 file path=usr/share/man/man3c/popen.3c
258 file path=usr/share/man/man3c/port_alert.3c
259 file path=usr/share/man/man3c/port_associate.3c
```

```
260 file path=usr/share/man/man3c/port_create.3c
261 file path=usr/share/man/man3c/port_get.3c
262 file path=usr/share/man/man3c/port_send.3c
263 file path=usr/share/man/man3c/posix_fadvise.3c
264 file path=usr/share/man/man3c/posix_fallocate.3c
265 file path=usr/share/man/man3c/posix_madvise.3c
266 file path=usr/share/man/man3c/posix_memalign.3c
267 file path=usr/share/man/man3c/posix_openpt.3c
268 file path=usr/share/man/man3c/posix_spawn.3c
269 file path=usr/share/man/man3c/posix_spawn_file_actions_addclose.3c
270 file path=usr/share/man/man3c/posix_spawn_file_actions_addclosefrom_np.3c
271 file path=usr/share/man/man3c/posix_spawn_file_actions_adddup2.3c
272 file path=usr/share/man/man3c/posix_spawn_file_actions_destroy.3c
273 file path=usr/share/man/man3c/posix_spawn_pipe_np.3c
274 file path=usr/share/man/man3c/posix_spawnattr_destroy.3c
275 file path=usr/share/man/man3c/posix_spawnattr_getflags.3c
276 file path=usr/share/man/man3c/posix_spawnattr_getpgroup.3c
277 file path=usr/share/man/man3c/posix_spawnattr_getschedparam.3c
278 file path=usr/share/man/man3c/posix_spawnattr_getschedpolicy.3c
279 file path=usr/share/man/man3c/posix_spawnattr_getsigdefault.3c
280 file path=usr/share/man/man3c/posix_spawnattr_getsigignore_np.3c
281 file path=usr/share/man/man3c/posix_spawnattr_getsigmask.3c
282 file path=usr/share/man/man3c/printf.3c
283 file path=usr/share/man/man3c/priv_addset.3c
284 file path=usr/share/man/man3c/priv_set.3c
285 file path=usr/share/man/man3c/priv_str_to_set.3c
286 file path=usr/share/man/man3c/pset_getloadavg.3c
287 file path=usr/share/man/man3c/psignal.3c
288 file path=usr/share/man/man3c/pthread_atfork.3c
289 file path=usr/share/man/man3c/pthread_attr_getdetachstate.3c
290 file path=usr/share/man/man3c/pthread_attr_getguardsize.3c
291 file path=usr/share/man/man3c/pthread_attr_getinheritsched.3c
292 file path=usr/share/man/man3c/pthread_attr_getschedparam.3c
293 file path=usr/share/man/man3c/pthread_attr_getschedpolicy.3c
294 file path=usr/share/man/man3c/pthread_attr_getscope.3c
295 file path=usr/share/man/man3c/pthread_attr_getstack.3c
296 file path=usr/share/man/man3c/pthread_attr_getstackaddr.3c
297 file path=usr/share/man/man3c/pthread_attr_getstacksize.3c
298 file path=usr/share/man/man3c/pthread_attr_init.3c
299 file path=usr/share/man/man3c/pthread_barrier_destroy.3c
300 file path=usr/share/man/man3c/pthread_barrier_wait.3c
301 file path=usr/share/man/man3c/pthread_barrierattr_destroy.3c
302 file path=usr/share/man/man3c/pthread_barrierattr_getpshared.3c
303 file path=usr/share/man/man3c/pthread_cancel.3c
304 file path=usr/share/man/man3c/pthread_cleanup_pop.3c
305 file path=usr/share/man/man3c/pthread_cleanup_push.3c
306 file path=usr/share/man/man3c/pthread_cond_init.3c
307 file path=usr/share/man/man3c/pthread_cond_signal.3c
308 file path=usr/share/man/man3c/pthread_cond_wait.3c
309 file path=usr/share/man/man3c/pthread_condattr_getclock.3c
310 file path=usr/share/man/man3c/pthread_condattr_getpshared.3c
311 file path=usr/share/man/man3c/pthread_condattr_init.3c
312 file path=usr/share/man/man3c/pthread_create.3c
313 file path=usr/share/man/man3c/pthread_detach.3c
314 file path=usr/share/man/man3c/pthread_equal.3c
315 file path=usr/share/man/man3c/pthread_exit.3c
316 file path=usr/share/man/man3c/pthread_getconcurrency.3c
317 file path=usr/share/man/man3c/pthread_getschedparam.3c
318 file path=usr/share/man/man3c/pthread_getspecific.3c
319 file path=usr/share/man/man3c/pthread_join.3c
320 file path=usr/share/man/man3c/pthread_key_create.3c
321 file path=usr/share/man/man3c/pthread_key_delete.3c
322 file path=usr/share/man/man3c/pthread_kill.3c
323 file path=usr/share/man/man3c/pthread_mutex_getprioceiling.3c
324 file path=usr/share/man/man3c/pthread_mutex_consistent.3c
325 file path=usr/share/man/man3c/pthread_mutex_init.3c
```



```
326 file path=usr/share/man/man3c/pthread_mutex_lock.3c
327 file path=usr/share/man/man3c/pthread_mutex_timedlock.3c
328 file path=usr/share/man/man3c/pthread_mutexattr_getprioceiling.3c
329 file path=usr/share/man/man3c/pthread_mutexattr_getprotocol.3c
330 file path=usr/share/man/man3c/pthread_mutexattr_getpshared.3c
331 file path=usr/share/man/man3c/pthread_mutexattr_getrobust.3c
332 file path=usr/share/man/man3c/pthread_mutexattr_gettype.3c
333 file path=usr/share/man/man3c/pthread_mutexattr_init.3c
334 file path=usr/share/man/man3c/pthread_once.3c
335 file path=usr/share/man/man3c/pthread_rwlock_init.3c
336 file path=usr/share/man/man3c/pthread_rwlock_rdlock.3c
337 file path=usr/share/man/man3c/pthread_rwlock_timedrdlock.3c
338 file path=usr/share/man/man3c/pthread_rwlock_timedwrlock.3c
339 file path=usr/share/man/man3c/pthread_rwlock_unlock.3c
340 file path=usr/share/man/man3c/pthread_rwlock_wrlock.3c
341 file path=usr/share/man/man3c/pthread_rwlockattr_getpshared.3c
342 file path=usr/share/man/man3c/pthread_rwlockattr_init.3c
343 file path=usr/share/man/man3c/pthread_self.3c
344 file path=usr/share/man/man3c/pthread_setcancelstate.3c
345 file path=usr/share/man/man3c/pthread_setcanceltype.3c
346 file path=usr/share/man/man3c/pthread_setschedprio.3c
347 file path=usr/share/man/man3c/pthread_sigmask.3c
348 file path=usr/share/man/man3c/pthread_spin_destroy.3c
349 file path=usr/share/man/man3c/pthread_spin_lock.3c
350 file path=usr/share/man/man3c/pthread_spin_unlock.3c
351 file path=usr/share/man/man3c/pthread_testcancel.3c
352 file path=usr/share/man/man3c/ptrace.3c
353 file path=usr/share/man/man3c/ptsname.3c
354 file path=usr/share/man/man3c/putenv.3c
355 file path=usr/share/man/man3c/putpwent.3c
356 file path=usr/share/man/man3c/puts.3c
357 file path=usr/share/man/man3c/putspent.3c
358 file path=usr/share/man/man3c/putws.3c
359 file path=usr/share/man/man3c/qsort.3c
360 file path=usr/share/man/man3c/raise.3c
361 file path=usr/share/man/man3c/rand.3c
362 file path=usr/share/man/man3c/random.3c
363 file path=usr/share/man/man3c/rctl_walk.3c
364 file path=usr/share/man/man3c/rctlblk_set_value.3c
365 file path=usr/share/man/man3c/re_comp.3c
366 file path=usr/share/man/man3c/readdir.3c
367 file path=usr/share/man/man3c/realpath.3c
368 file path=usr/share/man/man3c/reboot.3c
369 file path=usr/share/man/man3c/regcmp.3c
370 file path=usr/share/man/man3c/regcomp.3c
371 file path=usr/share/man/man3c/remove.3c
372 file path=usr/share/man/man3c/rewind.3c
373 file path=usr/share/man/man3c/rewinddir.3c
374 file path=usr/share/man/man3c/rwlock.3c
375 file path=usr/share/man/man3c/scandir.3c
376 file path=usr/share/man/man3c/scanf.3c
377 file path=usr/share/man/man3c/sched_get_priority_max.3c
378 file path=usr/share/man/man3c/sched_getparam.3c
379 file path=usr/share/man/man3c/sched_getscheduler.3c
380 file path=usr/share/man/man3c/sched_rr_get_interval.3c
381 file path=usr/share/man/man3c/sched_setparam.3c
382 file path=usr/share/man/man3c/sched_setscheduler.3c
383 file path=usr/share/man/man3c/sched_yield.3c
384 file path=usr/share/man/man3c/schedctl_init.3c
385 file path=usr/share/man/man3c/seekdir.3c
386 file path=usr/share/man/man3c/select.3c
387 file path=usr/share/man/man3c/sem_close.3c
388 file path=usr/share/man/man3c/sem_destroy.3c
389 file path=usr/share/man/man3c/sem_getvalue.3c
390 file path=usr/share/man/man3c/sem_init.3c
391 file path=usr/share/man/man3c/sem_open.3c
```

```
392 file path=usr/share/man/man3c/sem_post.3c
393 file path=usr/share/man/man3c/sem_timedwait.3c
394 file path=usr/share/man/man3c/sem_unlink.3c
395 file path=usr/share/man/man3c/sem_wait.3c
396 file path=usr/share/man/man3c/semaphore.3c
397 file path=usr/share/man/man3c/setbuf.3c
398 file path=usr/share/man/man3c/setbuffer.3c
399 file path=usr/share/man/man3c/setcat.3c
400 file path=usr/share/man/man3c/setenv.3c
401 file path=usr/share/man/man3c/setjmp.3c
402 file path=usr/share/man/man3c/setkey.3c
403 file path=usr/share/man/man3c/setlabel.3c
404 file path=usr/share/man/man3c/setlocale.3c
405 file path=usr/share/man/man3c/shm_open.3c
406 file path=usr/share/man/man3c/shm_unlink.3c
407 file path=usr/share/man/man3c/sigfpe.3c
408 file path=usr/share/man/man3c/siginterrupt.3c
409 file path=usr/share/man/man3c/signal.3c
410 file path=usr/share/man/man3c/signalfd.3c
411 file path=usr/share/man/man3c/sigqueue.3c
412 file path=usr/share/man/man3c/sigsetops.3c
413 file path=usr/share/man/man3c/sigstack.3c
414 file path=usr/share/man/man3c/sigwaitinfo.3c
415 file path=usr/share/man/man3c/sleep.3c
416 file path=usr/share/man/man3c/smt_pause.3c
417 file path=usr/share/man/man3c/ssignal.3c
418 file path=usr/share/man/man3c/stack_getbounds.3c
419 file path=usr/share/man/man3c/stack_inbounds.3c
420 file path=usr/share/man/man3c/stack_setbounds.3c
421 file path=usr/share/man/man3c/stack_violation.3c
422 file path=usr/share/man/man3c/stdio.3c
423 file path=usr/share/man/man3c/str2sig.3c
424 file path=usr/share/man/man3c/strcoll.3c
425 file path=usr/share/man/man3c/strerror.3c
426 file path=usr/share/man/man3c/strfmon.3c
427 file path=usr/share/man/man3c/strftime.3c
428 file path=usr/share/man/man3c/string.3c
429 file path=usr/share/man/man3c/string_to_decimal.3c
430 file path=usr/share/man/man3c/strptime.3c
431 file path=usr/share/man/man3c/strsignal.3c
432 file path=usr/share/man/man3c/strtod.3c
433 file path=usr/share/man/man3c/strtoimax.3c
434 file path=usr/share/man/man3c/strtol.3c
435 file path=usr/share/man/man3c/strtoul.3c
436 file path=usr/share/man/man3c/strtows.3c
437 file path=usr/share/man/man3c/strxfrm.3c
438 file path=usr/share/man/man3c/swab.3c
439 file path=usr/share/man/man3c/sync_instruction_memory.3c
440 file path=usr/share/man/man3c/sysconf.3c
441 file path=usr/share/man/man3c/syslog.3c
442 file path=usr/share/man/man3c/system.3c
443 file path=usr/share/man/man3c/tcdrain.3c
444 file path=usr/share/man/man3c/tcflow.3c
445 file path=usr/share/man/man3c/tcflush.3c
446 file path=usr/share/man/man3c/tcgetattr.3c
447 file path=usr/share/man/man3c/tcgetpgrp.3c
448 file path=usr/share/man/man3c/tcgetsid.3c
449 file path=usr/share/man/man3c/tcsendbreak.3c
450 file path=usr/share/man/man3c/tcsetattr.3c
451 file path=usr/share/man/man3c/tcsetpgrp.3c
452 file path=usr/share/man/man3c/tell.3c
453 file path=usr/share/man/man3c/telldir.3c
454 file path=usr/share/man/man3c/termios.3c
455 file path=usr/share/man/man3c/thr_create.3c
456 file path=usr/share/man/man3c/thr_exit.3c
457 file path=usr/share/man/man3c/thr_getconcurrency.3c
```



```
458 file path=usr/share/man/man3c/thr_getprio.3c
459 file path=usr/share/man/man3c/thr_join.3c
460 file path=usr/share/man/man3c/thr_keycreate.3c
461 file path=usr/share/man/man3c/thr_kill.3c
462 file path=usr/share/man/man3c/thr_main.3c
463 file path=usr/share/man/man3c/thr_min_stack.3c
464 file path=usr/share/man/man3c/thr_self.3c
465 file path=usr/share/man/man3c/thr_sigsetmask.3c
466 file path=usr/share/man/man3c/thr_stksegment.3c
467 file path=usr/share/man/man3c/thr_suspend.3c
468 file path=usr/share/man/man3c/thr_yield.3c
469 file path=usr/share/man/man3c/timer_create.3c
470 file path=usr/share/man/man3c/timer_delete.3c
471 file path=usr/share/man/man3c/timer_settime.3c
472 file path=usr/share/man/man3c/timeradd.3c
473 file path=usr/share/man/man3c/timerfd_create.3c
474 file path=usr/share/man/man3c/tmpfile.3c
475 file path=usr/share/man/man3c/tmpnam.3c
476 file path=usr/share/man/man3c/toascii.3c
477 file path=usr/share/man/man3c/tolower.3c
478 file path=usr/share/man/man3c/toupper.3c
479 file path=usr/share/man/man3c/towlower.3c
480 file path=usr/share/man/man3c/towupper.3c
481 file path=usr/share/man/man3c/truncate.3c
482 file path=usr/share/man/man3c/tsearch.3c
483 file path=usr/share/man/man3c/ttyname.3c
484 file path=usr/share/man/man3c/ttyslot.3c
485 file path=usr/share/man/man3c/u8_strcmp.3c
486 file path=usr/share/man/man3c/u8_textprep_str.3c
487 file path=usr/share/man/man3c/u8_validate.3c
488 file path=usr/share/man/man3c/ualarm.3c
489 file path=usr/share/man/man3c/uconv_u16tou32.3c
490 file path=usr/share/man/man3c/ucred.3c
491 file path=usr/share/man/man3c/ungetc.3c
492 file path=usr/share/man/man3c/ungetwc.3c
493 file path=usr/share/man/man3c/unlockpt.3c
494 file path=usr/share/man/man3c/unsetenv.3c
495 file path=usr/share/man/man3c/uselocale.3c
496 file path=usr/share/man/man3c/usleep.3c
497 file path=usr/share/man/man3c/vfwprintf.3c
498 file path=usr/share/man/man3c/vlfmt.3c
499 file path=usr/share/man/man3c/vpfmt.3c
500 file path=usr/share/man/man3c/vprintf.3c
501 file path=usr/share/man/man3c/vsyslog.3c
502 file path=usr/share/man/man3c/wait.3c
503 file path=usr/share/man/man3c/wait3.3c
504 file path=usr/share/man/man3c/waitpid.3c
505 file path=usr/share/man/man3c/walkcontext.3c
506 file path=usr/share/man/man3c/wcpcpy.3c
507 file path=usr/share/man/man3c/wcrtomb.3c
508 file path=usr/share/man/man3c/wcscasecmp.3c
509 file path=usr/share/man/man3c/wcscoll.3c
510 file path=usr/share/man/man3c/wcsdup.3c
511 file path=usr/share/man/man3c/wcsftime.3c
512 file path=usr/share/man/man3c/wcslen.3c
513 file path=usr/share/man/man3c/wcsrtombs.3c
514 file path=usr/share/man/man3c/wcsstr.3c
515 file path=usr/share/man/man3c/wcstod.3c
516 file path=usr/share/man/man3c/wcstoimax.3c
517 file path=usr/share/man/man3c/wcstol.3c
518 file path=usr/share/man/man3c/wcstoul.3c
519 file path=usr/share/man/man3c/wcstring.3c
520 file path=usr/share/man/man3c/wcswidth.3c
521 file path=usr/share/man/man3c/wcsxfrm.3c
522 file path=usr/share/man/man3c/wctob.3c
523 file path=usr/share/man/man3c/wctomb.3c
```

```
file path=usr/share/man/man3c/wctrans.3c
file path=usr/share/man/man3c/wctype.3c
file path=usr/share/man/man3c/wcwidth.3c
file path=usr/share/man/man3c/wmemchr.3c
file path=usr/share/man/man3c/wmemcmp.3c
file path=usr/share/man/man3c/wmemcpy.3c
file path=usr/share/man/man3c/wmemmove.3c
file path=usr/share/man/man3c/wmemset.3c
file path=usr/share/man/man3c/wordexp.3c
file path=usr/share/man/man3c/wsprintf.3c
file path=usr/share/man/man3c/wsscanf.3c
file path=usr/share/man/man3c/wstring.3c
link path=usr/share/man/man3c/FD_CLR.3c target=select.3c
link path=usr/share/man/man3c/FD_ISSET.3c target=select.3c
link path=usr/share/man/man3c/FD_SET.3c target=select.3c
link path=usr/share/man/man3c/FD_ZERO.3c target=select.3c
link path=usr/share/man/man3c/__flbf.3c target=__fbufsize.3c
link path=usr/share/man/man3c/__fpending.3c target=__fbufsize.3c
link path=usr/share/man/man3c/__fpurge.3c target=__fbufsize.3c
link path=usr/share/man/man3c/__freadable.3c target=__fbufsize.3c
link path=usr/share/man/man3c/__freading.3c target=__fbufsize.3c
link path=usr/share/man/man3c/__fsetlocking.3c target=__fbufsize.3c
link path=usr/share/man/man3c/__fwritable.3c target=__fbufsize.3c
link path=usr/share/man/man3c/__fwriting.3c target=__fbufsize.3c
link path=usr/share/man/man3c/_edata.3c target=end.3c
link path=usr/share/man/man3c/_end.3c target=end.3c
link path=usr/share/man/man3c/_etext.3c target=end.3c
link path=usr/share/man/man3c/_exithandle.3c target=exit.3c
link path=usr/share/man/man3c/_flushlbf.3c target=__fbufsize.3c
link path=usr/share/man/man3c/_setjmp.3c target=_longjmp.3c
link path=usr/share/man/man3c/addrtosymstr.3c target=walkcontext.3c
link path=usr/share/man/man3c/aiowrite.3c target=aioread.3c
link path=usr/share/man/man3c/alloca.3c target=malloc.3c
link path=usr/share/man/man3c/alphasort.3c target=scandir.3c
link path=usr/share/man/man3c/arc4random_buf.3c target=arc4random.3c
link path=usr/share/man/man3c/arc4random_uniform.3c target=arc4random.3c
link path=usr/share/man/man3c/ascftime.3c target=strftime.3c
link path=usr/share/man/man3c/asctime.3c target=ctime.3c
link path=usr/share/man/man3c/asctime_r.3c target=ctime.3c
link path=usr/share/man/man3c/asprintf.3c target=printf.3c
link path=usr/share/man/man3c/atof.3c target=strtod.3c
link path=usr/share/man/man3c/atoi.3c target=strtol.3c
link path=usr/share/man/man3c/atol.3c target=strtol.3c
link path=usr/share/man/man3c/atoll.3c target=strtol.3c
link path=usr/share/man/man3c/atomic_add_16.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_16_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_32.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_32_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_64.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_64_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_8.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_8_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_char.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_char_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_int.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_int_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_long.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_long_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_ptr.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_ptr_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_short.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_add_short_nv.3c target=atomic_add.3c
link path=usr/share/man/man3c/atomic_and_16.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_16_nv.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_32.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_32_nv.3c target=atomic_and.3c
```



```
link path=usr/share/man/man3c/atomic_and_64.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_64_nv.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_8.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_8_nv.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_uchar.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_uchar_nv.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_uint.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_uint_nv.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_ulong.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_ulong_nv.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_ushort.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_and_ushort_nv.3c target=atomic_and.3c
link path=usr/share/man/man3c/atomic_cas_16.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_32.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_64.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_8.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_ptr.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_uchar.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_uint.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_ulong.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_cas_ushort.3c target=atomic_cas.3c
link path=usr/share/man/man3c/atomic_clear_long_excl.3c target=atomic_bits.3c
link path=usr/share/man/man3c/atomic_dec_16.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_16_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_32.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_32_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_64.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_64_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_8.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_8_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_ptr.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_ptr_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_uchar.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_uchar_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_uint.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_uint_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_ulong.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_ulong_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_ushort.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_dec_ushort_nv.3c target=atomic_dec.3c
link path=usr/share/man/man3c/atomic_inc_16.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_16_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_32.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_32_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_64.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_64_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_8.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_8_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_ptr.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_ptr_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_uchar.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_uchar_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_uint.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_uint_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_ulong.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_ulong_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_ushort.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_inc_ushort_nv.3c target=atomic_inc.3c
link path=usr/share/man/man3c/atomic_or_16.3c target=atomic_or.3c
link path=usr/share/man/man3c/atomic_or_16_nv.3c target=atomic_or.3c
link path=usr/share/man/man3c/atomic_or_32.3c target=atomic_or.3c
link path=usr/share/man/man3c/atomic_or_32_nv.3c target=atomic_or.3c
link path=usr/share/man/man3c/atomic_or_64.3c target=atomic_or.3c
link path=usr/share/man/man3c/atomic_or_64_nv.3c target=atomic_or.3c
link path=usr/share/man/man3c/atomic_or_8.3c target=atomic_or.3c
link path=usr/share/man/man3c/atomic_or_8_nv.3c target=atomic_or.3c
```

```
656 link path=usr/share/man/man3c/atomic_or_uchar.3c target=atomic_or.3c
657 link path=usr/share/man/man3c/atomic_or_uchar_nv.3c target=atomic_or.3c
658 link path=usr/share/man/man3c/atomic_or_uint.3c target=atomic_or.3c
659 link path=usr/share/man/man3c/atomic_or_uint_nv.3c target=atomic_or.3c
660 link path=usr/share/man/man3c/atomic_or_ulong.3c target=atomic_or.3c
661 link path=usr/share/man/man3c/atomic_or_ulong_nv.3c target=atomic_or.3c
662 link path=usr/share/man/man3c/atomic_or_ushort.3c target=atomic_or.3c
663 link path=usr/share/man/man3c/atomic_or_ushort_nv.3c target=atomic_or.3c
664 link path=usr/share/man/man3c/atomic_set_long_excl.3c target=atomic_bits.3c
665 link path=usr/share/man/man3c/atomic_swap_16.3c target=atomic_swap.3c
666 link path=usr/share/man/man3c/atomic_swap_32.3c target=atomic_swap.3c
667 link path=usr/share/man/man3c/atomic_swap_64.3c target=atomic_swap.3c
668 link path=usr/share/man/man3c/atomic_swap_8.3c target=atomic_swap.3c
669 link path=usr/share/man/man3c/atomic_swap_ptr.3c target=atomic_swap.3c
670 link path=usr/share/man/man3c/atomic_swap_uchar.3c target=atomic_swap.3c
671 link path=usr/share/man/man3c/atomic_swap_uint.3c target=atomic_swap.3c
672 link path=usr/share/man/man3c/atomic_swap_ulong.3c target=atomic_swap.3c
673 link path=usr/share/man/man3c/atomic_swap_ushort.3c target=atomic_swap.3c
674 link path=usr/share/man/man3c/backtrace.3c target=walkcontext.3c
675 link path=usr/share/man/man3c/backtrace_symbols.3c target=walkcontext.3c
676 link path=usr/share/man/man3c/backtrace_symbols_fd.3c target=walkcontext.3c
677 link path=usr/share/man/man3c/bcmp.3c target=bstring.3c
678 link path=usr/share/man/man3c/bcopy.3c target=bstring.3c
679 link path=usr/share/man/man3c/bind_textdomain_codeset.3c target=gettext.3c
680 link path=usr/share/man/man3c/bindtextdomain.3c target=gettext.3c
681 link path=usr/share/man/man3c/btowc_l.3c target=btowc.3c
682 link path=usr/share/man/man3c/bzero.3c target=bstring.3c
683 link path=usr/share/man/man3c/calloc.3c target=malloc.3c
684 link path=usr/share/man/man3c/canonicalize_file_name.3c target=realpath.3c
685 link path=usr/share/man/man3c/catclose.3c target=catopen.3c
686 link path=usr/share/man/man3c/cfgetospeed.3c target=cfgetispeed.3c
687 link path=usr/share/man/man3c/cfsetospeed.3c target=cfsetispeed.3c
688 link path=usr/share/man/man3c/cftime.3c target=strftime.3c
689 link path=usr/share/man/man3c/clearerr.3c target=ferror.3c
690 link path=usr/share/man/man3c/clock_getres.3c target=clock_settime.3c
691 link path=usr/share/man/man3c/clock_gettime.3c target=clock_settime.3c
692 link path=usr/share/man/man3c/closelog.3c target=syslog.3c
693 link path=usr/share/man/man3c/cond_broadcast.3c target=cond_init.3c
694 link path=usr/share/man/man3c/cond_destroy.3c target=cond_init.3c
695 link path=usr/share/man/man3c/cond_reltimedwait.3c target=cond_init.3c
696 link path=usr/share/man/man3c/cond_signal.3c target=cond_init.3c
697 link path=usr/share/man/man3c/cond_timedwait.3c target=cond_init.3c
698 link path=usr/share/man/man3c/cond_wait.3c target=cond_init.3c
699 link path=usr/share/man/man3c/csetcol.3c target=cset.3c
700 link path=usr/share/man/man3c/csetlen.3c target=cset.3c
701 link path=usr/share/man/man3c/csetno.3c target=cset.3c
702 link path=usr/share/man/man3c/ctermid_r.3c target=ctermid.3c
703 link path=usr/share/man/man3c/ctime_r.3c target=ctime.3c
704 link path=usr/share/man/man3c/dbm_clearerr.3c target=ndbm.3c
705 link path=usr/share/man/man3c/dbm_close.3c target=ndbm.3c
706 link path=usr/share/man/man3c/dbm_delete.3c target=ndbm.3c
707 link path=usr/share/man/man3c/dbm_error.3c target=ndbm.3c
708 link path=usr/share/man/man3c/dbm_fetch.3c target=ndbm.3c
709 link path=usr/share/man/man3c/dbm_firstkey.3c target=ndbm.3c
710 link path=usr/share/man/man3c/dbm_nextkey.3c target=ndbm.3c
711 link path=usr/share/man/man3c/dbm_open.3c target=ndbm.3c
712 link path=usr/share/man/man3c/dbm_store.3c target=ndbm.3c
713 link path=usr/share/man/man3c/dcgettext.3c target=gettext.3c
714 link path=usr/share/man/man3c/dcngettext.3c target=gettext.3c
715 link path=usr/share/man/man3c/decimal_to_double.3c \
716     target=decimal_to_floating.3c
717 link path=usr/share/man/man3c/decimal_to_extended.3c \
718     target=decimal_to_floating.3c
719 link path=usr/share/man/man3c/decimal_to_quadruple.3c \
720     target=decimal_to_floating.3c
721 link path=usr/share/man/man3c/decimal_to_single.3c \
```



```
722     target=decimal_to_floating.3c
723 link path=usr/share/man/man3c/dgettext.3c target=gettext.3c
724 link path=usr/share/man/man3c/dladdr1.3c target=dladdr.3c
725 link path=usr/share/man/man3c/dlmopen.3c target=dlopen.3c
726 link path=usr/share/man/man3c/dngettext.3c target=gettext.3c
727 link path=usr/share/man/man3c/door_setparam.3c target=door_getparam.3c
728 link path=usr/share/man/man3c/door_unbind.3c target=door_bind.3c
729 link path=usr/share/man/man3c/double_to_decimal.3c \
730     target=floating_to_decimal.3c
731 link path=usr/share/man/man3c/dup3.3c target=dup2.3c
732 link path=usr/share/man/man3c/duplocale.3c target=newlocale.3c
733 link path=usr/share/man/man3c/edata.3c target=end.3c
734 link path=usr/share/man/man3c/endgrent.3c target=getgrnam.3c
735 link path=usr/share/man/man3c/endnetgrent.3c target=getnetgrent.3c
736 link path=usr/share/man/man3c/endpwent.3c target=getpwnam.3c
737 link path=usr/share/man/man3c/endspent.3c target=getspnam.3c
738 link path=usr/share/man/man3c/endusershell.3c target=getusershell.3c
739 link path=usr/share/man/man3c/endutent.3c target=getutent.3c
740 link path=usr/share/man/man3c/endutxent.3c target=getutxent.3c
741 link path=usr/share/man/man3c/epoll_create1.3c target=epoll_create.3c
742 link path=usr/share/man/man3c/epoll_pwait.3c target=epoll_wait.3c
743 link path=usr/share/man/man3c/erand48.3c target=drand48.3c
744 link path=usr/share/man/man3c/errno.3c target=perror.3c
745 link path=usr/share/man/man3c/errx.3c target=err.3c
746 link path=usr/share/man/man3c/etext.3c target=end.3c
747 link path=usr/share/man/man3c/euccol.3c target=euclen.3c
748 link path=usr/share/man/man3c/eucscol.3c target=euclen.3c
749 link path=usr/share/man/man3c/explicit_bzero.3c target=bstring.3c
750 link path=usr/share/man/man3c/extended_to_decimal.3c \
751     target=floating_to_decimal.3c
752 link path=usr/share/man/man3c/fconvert.3c target=econvert.3c
753 link path=usr/share/man/man3c/fcvt.3c target=ecvt.3c
754 link path=usr/share/man/man3c/fdopendir.3c target=opendir.3c
755 link path=usr/share/man/man3c/fdwalk.3c target=closefrom.3c
756 link path=usr/share/man/man3c/feof.3c target=ferror.3c
757 link path=usr/share/man/man3c/ffsl.3c target=ffs.3c
758 link path=usr/share/man/man3c/ffsll.3c target=ffs.3c
759 link path=usr/share/man/man3c/fgetgrent.3c target=getgrnam.3c
760 link path=usr/share/man/man3c/fgetgrent_r.3c target=getgrnam.3c
761 link path=usr/share/man/man3c/fgetpwent.3c target=getpwnam.3c
762 link path=usr/share/man/man3c/fgetpwent_r.3c target=getpwnam.3c
763 link path=usr/share/man/man3c/fgets.3c target=gets.3c
764 link path=usr/share/man/man3c/fgetspent.3c target=getspnam.3c
765 link path=usr/share/man/man3c/fgetspent_r.3c target=getspnam.3c
766 link path=usr/share/man/man3c/fgetwc_l.3c target=fgetwc.3c
767 link path=usr/share/man/man3c/fgetws.3c target=getws.3c
768 link path=usr/share/man/man3c/file_to_decimal.3c target=string_to_decimal.3c
769 link path=usr/share/man/man3c/fileno.3c target=ferror.3c
770 link path=usr/share/man/man3c/finite.3c target=isnand.3c
771 link path=usr/share/man/man3c/fls.3c target=ffs.3c
772 link path=usr/share/man/man3c/flsl.3c target=ffs.3c
773 link path=usr/share/man/man3c/flsll.3c target=ffs.3c
774 link path=usr/share/man/man3c/fpclass.3c target=isnand.3c
775 link path=usr/share/man/man3c/fpgetmask.3c target=fpgetround.3c
776 link path=usr/share/man/man3c/fpgetsticky.3c target=fpgetround.3c
777 link path=usr/share/man/man3c/fprintf.3c target=printf.3c
778 link path=usr/share/man/man3c/fpsetmask.3c target=fpgetround.3c
779 link path=usr/share/man/man3c/fpsetround.3c target=fpgetround.3c
780 link path=usr/share/man/man3c/fpsetsticky.3c target=fpgetround.3c
781 link path=usr/share/man/man3c/fputs.3c target=puts.3c
782 link path=usr/share/man/man3c/free.3c target=malloc.3c
783 link path=usr/share/man/man3c/freelocale.3c target=newlocale.3c
784 link path=usr/share/man/man3c/fscanf.3c target=scanf.3c
785 link path=usr/share/man/man3c/fseeko.3c target=fseek.3c
786 link path=usr/share/man/man3c/fsetattr.3c target=fgetattr.3c
787 link path=usr/share/man/man3c/ftello.3c target=ftell.3c
```

```
link path=usr/share/man/man3c/ftruncate.3c target=truncate.3c
link path=usr/share/man/man3c/ftrylockfile.3c target=flockfile.3c
link path=usr/share/man/man3c/func_to_decimal.3c target=string_to_decimal.3c
link path=usr/share/man/man3c/funlockfile.3c target=flockfile.3c
link path=usr/share/man/man3c/gconvert.3c target=econvert.3c
link path=usr/share/man/man3c/gcvt.3c target=ecvt.3c
link path=usr/share/man/man3c/getattrat.3c target=fgetattr.3c
link path=usr/share/man/man3c/getc.3c target=fgetc.3c
link path=usr/share/man/man3c/getc_unlocked.3c target=fgetc.3c
link path=usr/share/man/man3c/getchar.3c target=fgetc.3c
link path=usr/share/man/man3c/getchar_unlocked.3c target=fgetc.3c
link path=usr/share/man/man3c/getdelim.3c target=getline.3c
link path=usr/share/man/man3c/getextmntent.3c target=getmntent.3c
link path=usr/share/man/man3c/getgrent.3c target=getgrnam.3c
link path=usr/share/man/man3c/getgrent_r.3c target=getgrnam.3c
link path=usr/share/man/man3c/getgrgid.3c target=getgrnam.3c
link path=usr/share/man/man3c/getgrgid_r.3c target=getgrnam.3c
link path=usr/share/man/man3c/getgrnam_r.3c target=getgrnam.3c
link path=usr/share/man/man3c/gethomelgroup.3c target=getcpuid.3c
link path=usr/share/man/man3c/gethrvtime.3c target=gethrtime.3c
link path=usr/share/man/man3c/getlogin_r.3c target=getlogin.3c
link path=usr/share/man/man3c/getmntany.3c target=getmntent.3c
link path=usr/share/man/man3c/getnetgrent_r.3c target=getnetgrent.3c
link path=usr/share/man/man3c/get_nprocs_conf.3c target=get_nprocs.3c
link path=usr/share/man/man3c/getpassphrase.3c target=getpass.3c
link path=usr/share/man/man3c/getpwent.3c target=getpwnam.3c
link path=usr/share/man/man3c/getpwent_r.3c target=getpwnam.3c
link path=usr/share/man/man3c/getpwnam_r.3c target=getpwnam.3c
link path=usr/share/man/man3c/getpwuid.3c target=getpwnam.3c
link path=usr/share/man/man3c/getpwuid_r.3c target=getpwnam.3c
link path=usr/share/man/man3c/getspent.3c target=getspnam.3c
link path=usr/share/man/man3c/getspent_r.3c target=getspnam.3c
link path=usr/share/man/man3c/getspnam_r.3c target=getspnam.3c
link path=usr/share/man/man3c/getutid.3c target=getutent.3c
link path=usr/share/man/man3c/getutline.3c target=getutent.3c
link path=usr/share/man/man3c/getutmp.3c target=getutxent.3c
link path=usr/share/man/man3c/getutmpx.3c target=getutxent.3c
link path=usr/share/man/man3c/getutxid.3c target=getutxent.3c
link path=usr/share/man/man3c/getutxline.3c target=getutxent.3c
link path=usr/share/man/man3c/getvfsany.3c target=getvfsent.3c
link path=usr/share/man/man3c/getvfsfile.3c target=getvfsent.3c
link path=usr/share/man/man3c/getvfsspec.3c target=getvfsent.3c
link path=usr/share/man/man3c/getw.3c target=fgetc.3c
link path=usr/share/man/man3c/getwc_l.3c target=getwc.3c
link path=usr/share/man/man3c/getwchar_l.3c target=getwchar.3c
link path=usr/share/man/man3c/getzoneidbyname.3c target=getzoneid.3c
link path=usr/share/man/man3c/getzonenamebyid.3c target=getzoneid.3c
link path=usr/share/man/man3c/globfree.3c target=glob.3c
link path=usr/share/man/man3c/gmtime.3c target=ctime.3c
link path=usr/share/man/man3c/gmtime_r.3c target=ctime.3c
link path=usr/share/man/man3c/gsignal.3c target=ssignal.3c
link path=usr/share/man/man3c/hasmntopt.3c target=getmntent.3c
link path=usr/share/man/man3c/hcreate.3c target=hsearch.3c
link path=usr/share/man/man3c/hdestroy.3c target=hsearch.3c
link path=usr/share/man/man3c/initstate.3c target=random.3c
link path=usr/share/man/man3c/innetgr.3c target=getnetgrent.3c
link path=usr/share/man/man3c/isalnum.3c target=ctype.3c
link path=usr/share/man/man3c/isalnum_l.3c target=ctype.3c
link path=usr/share/man/man3c/isalpha.3c target=ctype.3c
link path=usr/share/man/man3c/isalpha_l.3c target=ctype.3c
link path=usr/share/man/man3c/isascii.3c target=ctype.3c
link path=usr/share/man/man3c/isblank.3c target=ctype.3c
link path=usr/share/man/man3c/isblank_l.3c target=ctype.3c
link path=usr/share/man/man3c/iscntrl.3c target=ctype.3c
link path=usr/share/man/man3c/iscntrl_l.3c target=ctype.3c
link path=usr/share/man/man3c/isdigit.3c target=ctype.3c
```



```
link path=usr/share/man/man3c/isdigit_l.3c target=ctype.3c
link path=usr/share/man/man3c/isenglish.3c target=iswalpha.3c
link path=usr/share/man/man3c/isgraph.3c target=ctype.3c
link path=usr/share/man/man3c/isgraph_l.3c target=ctype.3c
link path=usr/share/man/man3c/isideogram.3c target=iswalpha.3c
link path=usr/share/man/man3c/islower.3c target=ctype.3c
link path=usr/share/man/man3c/islower_l.3c target=ctype.3c
link path=usr/share/man/man3c/isnanf.3c target=isnand.3c
link path=usr/share/man/man3c/isnumber.3c target=iswalpha.3c
link path=usr/share/man/man3c/isphonogram.3c target=iswalpha.3c
link path=usr/share/man/man3c/isprint.3c target=ctype.3c
link path=usr/share/man/man3c/isprint_l.3c target=ctype.3c
link path=usr/share/man/man3c/ispunct.3c target=ctype.3c
link path=usr/share/man/man3c/ispunct_l.3c target=ctype.3c
link path=usr/share/man/man3c/isspace.3c target=ctype.3c
link path=usr/share/man/man3c/isspace_l.3c target=ctype.3c
link path=usr/share/man/man3c/isspecial.3c target=iswalpha.3c
link path=usr/share/man/man3c/isupper.3c target=ctype.3c
link path=usr/share/man/man3c/isupper_l.3c target=ctype.3c
link path=usr/share/man/man3c/iswalnum.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswalnum_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswalpha_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswascii.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswblank.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswblank_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswcntrl.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswcntrl_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswctype_l.3c target=iswctype.3c
link path=usr/share/man/man3c/iswdigit.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswdigit_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswgraph.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswgraph_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswhexnumber.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswhexnumber_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswideogram.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswideogram_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswlower.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswlower_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswnumber.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswnumber_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswphonogram.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswphonogram_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswprint.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswprint_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswpunct.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswpunct_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswspace.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswspace_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswspecial.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswspecial_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswupper.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswupper_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswxdigit.3c target=iswalpha.3c
link path=usr/share/man/man3c/iswxdigit_l.3c target=iswalpha.3c
link path=usr/share/man/man3c/isxdigit.3c target=ctype.3c
link path=usr/share/man/man3c/isxdigit_l.3c target=ctype.3c
link path=usr/share/man/man3c/jrand48.3c target=drand48.3c
link path=usr/share/man/man3c/l64a.3c target=a64l.3c
link path=usr/share/man/man3c/labs.3c target=abs.3c
link path=usr/share/man/man3c/lcong48.3c target=drand48.3c
link path=usr/share/man/man3c/ldiv.3c target=div.3c
link path=usr/share/man/man3c/lfind.3c target=lsearch.3c
link path=usr/share/man/man3c/llabs.3c target=abs.3c
link path=usr/share/man/man3c/lldiv.3c target=div.3c
link path=usr/share/man/man3c/lltostr.3c target=strtol.3c
link path=usr/share/man/man3c/localtime.3c target=ctime.3c
```

```
link path=usr/share/man/man3c/localtime_r.3c target=ctime.3c
link path=usr/share/man/man3c/longjmp.3c target=setjmp.3c
link path=usr/share/man/man3c/lrand48.3c target=drand48.3c
link path=usr/share/man/man3c/major.3c target=makedev.3c
link path=usr/share/man/man3c/mblen_l.3c target=mblen.3c
link path=usr/share/man/man3c/mbrlen_l.3c target=mbrlen.3c
link path=usr/share/man/man3c/mbrtowc_l.3c target=mbrtowc.3c
link path=usr/share/man/man3c/mbsinit_l.3c target=mbsinit.3c
link path=usr/share/man/man3c/mbsnrtowcs.3c target=mbsrtowcs.3c
link path=usr/share/man/man3c/mbsnrtowcs_l.3c target=mbsrtowcs.3c
link path=usr/share/man/man3c/mbsrtowcs_l.3c target=mbsrtowcs.3c
link path=usr/share/man/man3c/mbstowcs.3c target=mbsrtowcs.3c
link path=usr/share/man/man3c/mbstowcs_l.3c target=mbsrtowcs.3c
link path=usr/share/man/man3c/mbtowc_l.3c target=mbtowc.3c
link path=usr/share/man/man3c/memalign.3c target=malloc.3c
link path=usr/share/man/man3c/membar_consumer.3c target=membar_ops.3c
link path=usr/share/man/man3c/membar_enter.3c target=membar_ops.3c
link path=usr/share/man/man3c/membar_exit.3c target=membar_ops.3c
link path=usr/share/man/man3c/membar_producer.3c target=membar_ops.3c
link path=usr/share/man/man3c/memccpy.3c target=memory.3c
link path=usr/share/man/man3c/memchr.3c target=memory.3c
link path=usr/share/man/man3c/memcmp.3c target=memory.3c
link path=usr/share/man/man3c/memcpy.3c target=memory.3c
link path=usr/share/man/man3c/memmem.3c target=memory.3c
link path=usr/share/man/man3c/memmove.3c target=memory.3c
link path=usr/share/man/man3c/memset.3c target=memory.3c
link path=usr/share/man/man3c/minor.3c target=makedev.3c
link path=usr/share/man/man3c/mkfifoat.3c target=mkfifo.3c
link path=usr/share/man/man3c/mkdtemp.3c target=mkstemp.3c
link path=usr/share/man/man3c/mkostemp.3c target=mkstemp.3c
link path=usr/share/man/man3c/mkostemps.3c target=mkstemp.3c
link path=usr/share/man/man3c/mkstemps.3c target=mkstemp.3c
link path=usr/share/man/man3c/mq_reltimedreceive_np.3c target=mq_receive.3c
link path=usr/share/man/man3c/mq_reltimedsend_np.3c target=mq_send.3c
link path=usr/share/man/man3c/mq_timedreceive.3c target=mq_receive.3c
link path=usr/share/man/man3c/mq_timedsend.3c target=mq_send.3c
link path=usr/share/man/man3c/mrand48.3c target=drand48.3c
link path=usr/share/man/man3c/munlock.3c target=mlock.3c
link path=usr/share/man/man3c/munlockall.3c target=mlockall.3c
link path=usr/share/man/man3c/mutex_consistent.3c target=mutex_init.3c
link path=usr/share/man/man3c/mutex_destroy.3c target=mutex_init.3c
link path=usr/share/man/man3c/mutex_lock.3c target=mutex_init.3c
link path=usr/share/man/man3c/mutex_trylock.3c target=mutex_init.3c
link path=usr/share/man/man3c/mutex_unlock.3c target=mutex_init.3c
link path=usr/share/man/man3c/nftw.3c target=ftw.3c
link path=usr/share/man/man3c/ngettext.3c target=gettext.3c
link path=usr/share/man/man3c/nl_langinfo_l.3c target=nl_langinfo.3c
link path=usr/share/man/man3c/nrand48.3c target=drand48.3c
link path=usr/share/man/man3c/openlog.3c target=syslog.3c
link path=usr/share/man/man3c/pclose.3c target=popen.3c
link path=usr/share/man/man3c/port_dissociate.3c target=port_associate.3c
link path=usr/share/man/man3c/port_getn.3c target=port_get.3c
link path=usr/share/man/man3c/port_sendn.3c target=port_send.3c
link path=usr/share/man/man3c/posix_spawn_file_actions_addopen.3c \
    target=posix_spawn_file_actions_addclose.3c
link path=usr/share/man/man3c/posix_spawn_file_actions_init.3c \
    target=posix_spawn_file_actions_destroy.3c
link path=usr/share/man/man3c/posix_spawnattr_init.3c \
    target=posix_spawnattr_destroy.3c
link path=usr/share/man/man3c/posix_spawnattr_setflags.3c \
    target=posix_spawnattr_getflags.3c
link path=usr/share/man/man3c/posix_spawnattr_setpgroup.3c \
    target=posix_spawnattr_getpgroup.3c
link path=usr/share/man/man3c/posix_spawnattr_setschedparam.3c \
    target=posix_spawnattr_getschedparam.3c
link path=usr/share/man/man3c/posix_spawnattr_setschedpolicy.3c \
```



```
    target=posix_spawnattr_getschedpolicy.3c
link path=usr/share/man/man3c/posix_spawnattr_setsigdefault.3c \
    target=posix_spawnattr_getsigdefault.3c
link path=usr/share/man/man3c/posix_spawnattr_setsigignore_np.3c \
    target=posix_spawnattr_getsigignore_np.3c
link path=usr/share/man/man3c/posix_spawnattr_setsigmask.3c \
    target=posix_spawnattr_getsigmask.3c
link path=usr/share/man/man3c/posix_spawnp.3c target=posix_spawn.3c
link path=usr/share/man/man3c/printstack.3c target=walkcontext.3c
link path=usr/share/man/man3c/priv_allocset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_basicset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_copyset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_defaultset.3c target=priv_addset.3c
#endif /* ! codereview */
link path=usr/share/man/man3c/priv_delset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_emptyset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_fillset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_freeset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_getbyname.3c target=priv_str_to_set.3c
link path=usr/share/man/man3c/priv_getbynum.3c target=priv_str_to_set.3c
link path=usr/share/man/man3c/priv_getsetbyname.3c target=priv_str_to_set.3c
link path=usr/share/man/man3c/priv_getsetbynum.3c target=priv_str_to_set.3c
link path=usr/share/man/man3c/priv_gettext.3c target=priv_str_to_set.3c
link path=usr/share/man/man3c/priv_ineffect.3c target=priv_set.3c
link path=usr/share/man/man3c/priv_intersect.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_inverse.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_isemptyset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_isequalset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_isfullset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_ismember.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_issubset.3c target=priv_addset.3c
link path=usr/share/man/man3c/priv_set_to_str.3c target=priv_str_to_set.3c
link path=usr/share/man/man3c/priv_union.3c target=priv_addset.3c
link path=usr/share/man/man3c/pselect.3c target=select.3c
link path=usr/share/man/man3c/psiginfo.3c target=psignal.3c
link path=usr/share/man/man3c/pthread_attr_destroy.3c \
    target=pthread_attr_init.3c
link path=usr/share/man/man3c/pthread_attr_setdetachstate.3c \
    target=pthread_attr_getdetachstate.3c
link path=usr/share/man/man3c/pthread_attr_setguardsize.3c \
    target=pthread_attr_getguardsize.3c
link path=usr/share/man/man3c/pthread_attr_setinheritsched.3c \
    target=pthread_attr_getinheritsched.3c
link path=usr/share/man/man3c/pthread_attr_setschedparam.3c \
    target=pthread_attr_getschedparam.3c
link path=usr/share/man/man3c/pthread_attr_setschedpolicy.3c \
    target=pthread_attr_getschedpolicy.3c
link path=usr/share/man/man3c/pthread_attr_setscope.3c \
    target=pthread_attr_getscope.3c
link path=usr/share/man/man3c/pthread_attr_setstack.3c \
    target=pthread_attr_getstack.3c
link path=usr/share/man/man3c/pthread_attr_setstackaddr.3c \
    target=pthread_attr_getstackaddr.3c
link path=usr/share/man/man3c/pthread_attr_setstacksize.3c \
    target=pthread_attr_getstacksize.3c
link path=usr/share/man/man3c/pthread_barrier_init.3c \
    target=pthread_barrier_destroy.3c
link path=usr/share/man/man3c/pthread_barrierattr_init.3c \
    target=pthread_barrierattr_destroy.3c
link path=usr/share/man/man3c/pthread_barrierattr_setpshared.3c \
    target=pthread_barrierattr_getpshared.3c
link path=usr/share/man/man3c/pthread_cond_broadcast.3c \
    target=pthread_cond_signal.3c
link path=usr/share/man/man3c/pthread_cond_destroy.3c \
    target=pthread_cond_init.3c
link path=usr/share/man/man3c/pthread_cond_reltimedwait_np.3c \
```

```
1052     target=pthread_cond_wait.3c
1053 link path=usr/share/man/man3c/pthread_cond_timedwait.3c \
1054     target=pthread_cond_wait.3c
1055 link path=usr/share/man/man3c/pthread_condattr_destroy.3c \
1056     target=pthread_condattr_init.3c
1057 link path=usr/share/man/man3c/pthread_condattr_setclock.3c \
1058     target=pthread_condattr_getclock.3c
1059 link path=usr/share/man/man3c/pthread_condattr_setpshared.3c \
1060     target=pthread_condattr_getpshared.3c
1061 link path=usr/share/man/man3c/pthread_key_create_once_np.3c \
1062     target=pthread_key_create.3c
1063 link path=usr/share/man/man3c/pthread_mutex_destroy.3c \
1064     target=pthread_mutex_init.3c
1065 link path=usr/share/man/man3c/pthread_mutex_reltimedlock_np.3c \
1066     target=pthread_mutex_timedlock.3c
1067 link path=usr/share/man/man3c/pthread_mutex_setprioceiling.3c \
1068     target=pthread_mutex_getprioceiling.3c
1069 link path=usr/share/man/man3c/pthread_mutex_trylock.3c \
1070     target=pthread_mutex_lock.3c
1071 link path=usr/share/man/man3c/pthread_mutex_unlock.3c \
1072     target=pthread_mutex_lock.3c
1073 link path=usr/share/man/man3c/pthread_mutexattr_destroy.3c \
1074     target=pthread_mutexattr_init.3c
1075 link path=usr/share/man/man3c/pthread_mutexattr_setprioceiling.3c \
1076     target=pthread_mutexattr_getprioceiling.3c
1077 link path=usr/share/man/man3c/pthread_mutexattr_setprotocol.3c \
1078     target=pthread_mutexattr_getprotocol.3c
1079 link path=usr/share/man/man3c/pthread_mutexattr_setpshared.3c \
1080     target=pthread_mutexattr_getpshared.3c
1081 link path=usr/share/man/man3c/pthread_mutexattr_setrobust.3c \
1082     target=pthread_mutexattr_getrobust.3c
1083 link path=usr/share/man/man3c/pthread_mutexattr_settype.3c \
1084     target=pthread_mutexattr_gettype.3c
1085 link path=usr/share/man/man3c/pthread_rwlock_destroy.3c \
1086     target=pthread_rwlock_init.3c
1087 link path=usr/share/man/man3c/pthread_rwlock_reltimedrdlock_np.3c \
1088     target=pthread_rwlock_timedrdlock.3c
1089 link path=usr/share/man/man3c/pthread_rwlock_reltimedwrlock_np.3c \
1090     target=pthread_rwlock_timedwrlock.3c
1091 link path=usr/share/man/man3c/pthread_rwlock_tryrdlock.3c \
1092     target=pthread_rwlock_rdlock.3c
1093 link path=usr/share/man/man3c/pthread_rwlock_trywrlock.3c \
1094     target=pthread_rwlock_wrlock.3c
1095 link path=usr/share/man/man3c/pthread_rwlockattr_destroy.3c \
1096     target=pthread_rwlockattr_init.3c
1097 link path=usr/share/man/man3c/pthread_rwlockattr_setpshared.3c \
1098     target=pthread_rwlockattr_getpshared.3c
1099 link path=usr/share/man/man3c/pthread_setconcurrency.3c \
1100     target=pthread_getconcurrency.3c
1101 link path=usr/share/man/man3c/pthread_setschedparam.3c \
1102     target=pthread_getschedparam.3c
1103 link path=usr/share/man/man3c/pthread_setspecific.3c \
1104     target=pthread_getspecific.3c
1105 link path=usr/share/man/man3c/pthread_spin_init.3c \
1106     target=pthread_spin_destroy.3c
1107 link path=usr/share/man/man3c/pthread_spin_trylock.3c \
1108     target=pthread_spin_lock.3c
1109 link path=usr/share/man/man3c/putc.3c target=fputc.3c
1110 link path=usr/share/man/man3c/putc_unlocked.3c target=fputc.3c
1111 link path=usr/share/man/man3c/putchar.3c target=fputc.3c
1112 link path=usr/share/man/man3c/putchar_unlocked.3c target=fputc.3c
1113 link path=usr/share/man/man3c/putmntent.3c target=getmntent.3c
1114 link path=usr/share/man/man3c/pututline.3c target=getutent.3c
1115 link path=usr/share/man/man3c/pututxline.3c target=getutxent.3c
1116 link path=usr/share/man/man3c/putw.3c target=fputc.3c
1117 link path=usr/share/man/man3c/putwc.3c target=fputwc.3c
```



```
1118 link path=usr/share/man/man3c/putwchar.3c target=fputwc.3c
1119 link path=usr/share/man/man3c/qeconvert.3c target=econvert.3c
1120 link path=usr/share/man/man3c/qfconvert.3c target=econvert.3c
1121 link path=usr/share/man/man3c/qgconvert.3c target=econvert.3c
1122 link path=usr/share/man/man3c/quadruple_to_decimal.3c \
1123     target=floating_to_decimal.3c
1124 link path=usr/share/man/man3c/rand_r.3c target=rand.3c
1125 link path=usr/share/man/man3c/rctlblk_get_enforced_value.3c \
1126     target=rctlblk_set_value.3c
1127 link path=usr/share/man/man3c/rctlblk_get_firing_time.3c \
1128     target=rctlblk_set_value.3c
1129 link path=usr/share/man/man3c/rctlblk_get_global_action.3c \
1130     target=rctlblk_set_value.3c
1131 link path=usr/share/man/man3c/rctlblk_get_global_flags.3c \
1132     target=rctlblk_set_value.3c
1133 link path=usr/share/man/man3c/rctlblk_get_local_action.3c \
1134     target=rctlblk_set_value.3c
1135 link path=usr/share/man/man3c/rctlblk_get_local_flags.3c \
1136     target=rctlblk_set_value.3c
1137 link path=usr/share/man/man3c/rctlblk_get_privilege.3c \
1138     target=rctlblk_set_value.3c
1139 link path=usr/share/man/man3c/rctlblk_get_recipient_pid.3c \
1140     target=rctlblk_set_value.3c
1141 link path=usr/share/man/man3c/rctlblk_get_value.3c target=rctlblk_set_value.3c
1142 link path=usr/share/man/man3c/rctlblk_set_local_action.3c \
1143     target=rctlblk_set_value.3c
1144 link path=usr/share/man/man3c/rctlblk_set_local_flags.3c \
1145     target=rctlblk_set_value.3c
1146 link path=usr/share/man/man3c/rctlblk_set_privilege.3c \
1147     target=rctlblk_set_value.3c
1148 link path=usr/share/man/man3c/rctlblk_set_recipient_pid.3c \
1149     target=rctlblk_set_value.3c
1150 link path=usr/share/man/man3c/rctlblk_size.3c target=rctlblk_set_value.3c
1151 link path=usr/share/man/man3c/re_exec.3c target=re_comp.3c
1152 link path=usr/share/man/man3c/readdir_r.3c target=readdir.3c
1153 link path=usr/share/man/man3c/realloc.3c target=malloc.3c
1154 link path=usr/share/man/man3c/regerror.3c target=regcomp.3c
1155 link path=usr/share/man/man3c/regex.3c target=regcmp.3c
1156 link path=usr/share/man/man3c/regexec.3c target=regcomp.3c
1157 link path=usr/share/man/man3c/regfree.3c target=regcomp.3c
1158 link path=usr/share/man/man3c/remque.3c target=insque.3c
1159 link path=usr/share/man/man3c/resetmnttab.3c target=getmntent.3c
1160 link path=usr/share/man/man3c/rindex.3c target=index.3c
1161 link path=usr/share/man/man3c/rw_rdlock.3c target=rwlock.3c
1162 link path=usr/share/man/man3c/rw_tryrdlock.3c target=rwlock.3c
1163 link path=usr/share/man/man3c/rw_trywrlock.3c target=rwlock.3c
1164 link path=usr/share/man/man3c/rw_unlock.3c target=rwlock.3c
1165 link path=usr/share/man/man3c/rw_wrlock.3c target=rwlock.3c
1166 link path=usr/share/man/man3c/rwlock_destroy.3c target=rwlock.3c
1167 link path=usr/share/man/man3c/rwlock_init.3c target=rwlock.3c
1168 link path=usr/share/man/man3c/sched_get_priority_min.3c \
1169     target=sched_get_priority_max.3c
1170 link path=usr/share/man/man3c/schedctl_exit.3c target=schedctl_init.3c
1171 link path=usr/share/man/man3c/schedctl_lookup.3c target=schedctl_init.3c
1172 link path=usr/share/man/man3c/schedctl_start.3c target=schedctl_init.3c
1173 link path=usr/share/man/man3c/schedctl_stop.3c target=schedctl_init.3c
1174 link path=usr/share/man/man3c/seconvert.3c target=econvert.3c
1175 link path=usr/share/man/man3c/seed48.3c target=drand48.3c
1176 link path=usr/share/man/man3c/sem_reltimedwait_np.3c target=sem_timedwait.3c
1177 link path=usr/share/man/man3c/sem_trywait.3c target=sem_wait.3c
1178 link path=usr/share/man/man3c/sema_destroy.3c target=semaphore.3c
1179 link path=usr/share/man/man3c/sema_init.3c target=semaphore.3c
1180 link path=usr/share/man/man3c/sema_post.3c target=semaphore.3c
1181 link path=usr/share/man/man3c/sema_trywait.3c target=semaphore.3c
1182 link path=usr/share/man/man3c/sema_wait.3c target=semaphore.3c
1183 link path=usr/share/man/man3c/setattrat.3c target=fgetattr.3c
```

```
1184 link path=usr/share/man/man3c/setgrent.3c target=getgrnam.3c
1185 link path=usr/share/man/man3c/sethostname.3c target=gethostname.3c
1186 link path=usr/share/man/man3c/setlinebuf.3c target=setbuffer.3c
1187 link path=usr/share/man/man3c/setlogmask.3c target=syslog.3c
1188 link path=usr/share/man/man3c/setnetgrent.3c target=getnetgrent.3c
1189 link path=usr/share/man/man3c/setpriority.3c target=getpriority.3c
1190 link path=usr/share/man/man3c/setprogname.3c target=getprogname.3c
1191 link path=usr/share/man/man3c/setpwent.3c target=getpwnam.3c
1192 link path=usr/share/man/man3c/setspent.3c target=getspnam.3c
1193 link path=usr/share/man/man3c/setstate.3c target=random.3c
1194 link path=usr/share/man/man3c/settimeofday.3c target=gettimeofday.3c
1195 link path=usr/share/man/man3c/setusershell.3c target=getusershell.3c
1196 link path=usr/share/man/man3c/setutent.3c target=getutent.3c
1197 link path=usr/share/man/man3c/setutxent.3c target=getutxent.3c
1198 link path=usr/share/man/man3c/setvbuf.3c target=setbuf.3c
1199 link path=usr/share/man/man3c/sfconvert.3c target=econvert.3c
1200 link path=usr/share/man/man3c/sgconvert.3c target=econvert.3c
1201 link path=usr/share/man/man3c/sig2str.3c target=str2sig.3c
1202 link path=usr/share/man/man3c/sigaddset.3c target=sigsetops.3c
1203 link path=usr/share/man/man3c/sigdelset.3c target=sigsetops.3c
1204 link path=usr/share/man/man3c/sigemptyset.3c target=sigsetops.3c
1205 link path=usr/share/man/man3c/sigfillset.3c target=sigsetops.3c
1206 link path=usr/share/man/man3c/sighold.3c target=signal.3c
1207 link path=usr/share/man/man3c/sigignore.3c target=signal.3c
1208 link path=usr/share/man/man3c/sigismember.3c target=sigsetops.3c
1209 link path=usr/share/man/man3c/siglongjmp.3c target=setjmp.3c
1210 link path=usr/share/man/man3c/sigpause.3c target=signal.3c
1211 link path=usr/share/man/man3c/sigrelse.3c target=signal.3c
1212 link path=usr/share/man/man3c/sigset.3c target=signal.3c
1213 link path=usr/share/man/man3c/sigsetjmp.3c target=setjmp.3c
1214 link path=usr/share/man/man3c/sigtimedwait.3c target=sigwaitinfo.3c
1215 link path=usr/share/man/man3c/single_to_decimal.3c \
1216     target=floating_to_decimal.3c
1217 link path=usr/share/man/man3c/snprintf.3c target=printf.3c
1218 link path=usr/share/man/man3c/sprintf.3c target=printf.3c
1219 link path=usr/share/man/man3c/srand.3c target=rand.3c
1220 link path=usr/share/man/man3c/srand48.3c target=drand48.3c
1221 link path=usr/share/man/man3c/srandom.3c target=random.3c
1222 link path=usr/share/man/man3c/sscanf.3c target=scanf.3c
1223 link path=usr/share/man/man3c/stderr.3c target=stdio.3c
1224 link path=usr/share/man/man3c/stdin.3c target=stdio.3c
1225 link path=usr/share/man/man3c/stdout.3c target=stdio.3c
1226 link path=usr/share/man/man3c/stpcpy.3c target=string.3c
1227 link path=usr/share/man/man3c/stpncpy.3c target=string.3c
1228 link path=usr/share/man/man3c/strcasecmp.3c target=string.3c
1229 link path=usr/share/man/man3c/strcasecmp_l.3c target=string.3c
1230 link path=usr/share/man/man3c/strcasestr.3c target=string.3c
1231 link path=usr/share/man/man3c/strcasestr_l.3c target=string.3c
1232 link path=usr/share/man/man3c/strcat.3c target=string.3c
1233 link path=usr/share/man/man3c/strchr.3c target=string.3c
1234 link path=usr/share/man/man3c/strchrnul.3c target=string.3c
1235 link path=usr/share/man/man3c/strcmp.3c target=string.3c
1236 link path=usr/share/man/man3c/strcoll_l.3c target=strcoll.3c
1237 link path=usr/share/man/man3c/strcpy.3c target=string.3c
1238 link path=usr/share/man/man3c/strcspn.3c target=string.3c
1239 link path=usr/share/man/man3c/strdup.3c target=string.3c
1240 link path=usr/share/man/man3c/strdupa.3c target=string.3c
1241 link path=usr/share/man/man3c/strerror_l.3c target=strerror.3c
1242 link path=usr/share/man/man3c/strerror_r.3c target=strerror.3c
1243 link path=usr/share/man/man3c/strfmon_l.3c target=strfmon.3c
1244 link path=usr/share/man/man3c/strftime_l.3c target=strftime.3c
1245 link path=usr/share/man/man3c/strlcat.3c target=string.3c
1246 link path=usr/share/man/man3c/strlcpy.3c target=string.3c
1247 link path=usr/share/man/man3c/strlen.3c target=string.3c
1248 link path=usr/share/man/man3c/strncasecmp.3c target=string.3c
1249 link path=usr/share/man/man3c/strncasecmp_l.3c target=string.3c
```



```
1250 link path=usr/share/man/man3c/strncat.3c target=string.3c
1251 link path=usr/share/man/man3c/strncmp.3c target=string.3c
1252 link path=usr/share/man/man3c/strncpy.3c target=string.3c
1253 link path=usr/share/man/man3c/strndup.3c target=string.3c
1254 link path=usr/share/man/man3c/strndupa.3c target=string.3c
1255 link path=usr/share/man/man3c/strnlen.3c target=string.3c
1256 link path=usr/share/man/man3c/strnstr.3c target=string.3c
1257 link path=usr/share/man/man3c/strpbrk.3c target=string.3c
1258 link path=usr/share/man/man3c/strptime_l.3c target=strptime.3c
1259 link path=usr/share/man/man3c/strrchr.3c target=string.3c
1260 link path=usr/share/man/man3c/strsep.3c target=string.3c
1261 link path=usr/share/man/man3c/strspn.3c target=string.3c
1262 link path=usr/share/man/man3c/strstr.3c target=string.3c
1263 link path=usr/share/man/man3c/strtof.3c target=strtod.3c
1264 link path=usr/share/man/man3c/strtok.3c target=string.3c
1265 link path=usr/share/man/man3c/strtok_r.3c target=string.3c
1266 link path=usr/share/man/man3c/strtold.3c target=strtod.3c
1267 link path=usr/share/man/man3c/strtoll.3c target=strtol.3c
1268 link path=usr/share/man/man3c/strtoull.3c target=strtoul.3c
1269 link path=usr/share/man/man3c/strtoumax.3c target=strtoimax.3c
1270 link path=usr/share/man/man3c/strxfrm_l.3c target=strxfrm.3c
1271 link path=usr/share/man/man3c/swapcontext.3c target=makecontext.3c
1272 link path=usr/share/man/man3c/swprintf.3c target=fwprintf.3c
1273 link path=usr/share/man/man3c/swscanf.3c target=fwscanf.3c
1274 link path=usr/share/man/man3c/tdelete.3c target=tsearch.3c
1275 link path=usr/share/man/man3c/tempnam.3c target=tmpnam.3c
1276 link path=usr/share/man/man3c/textdomain.3c target=gettext.3c
1277 link path=usr/share/man/man3c/tfind.3c target=tsearch.3c
1278 link path=usr/share/man/man3c/thr_continue.3c target=thr_suspend.3c
1279 link path=usr/share/man/man3c/thr_getspecific.3c target=thr_keycreate.3c
1280 link path=usr/share/man/man3c/thr_keycreate_once.3c target=thr_keycreate.3c
1281 link path=usr/share/man/man3c/thr_setconcurrency.3c \
1282     target=thr_getconcurrency.3c
1283 link path=usr/share/man/man3c/thr_setprio.3c target=thr_getprio.3c
1284 link path=usr/share/man/man3c/thr_setspecific.3c target=thr_keycreate.3c
1285 link path=usr/share/man/man3c/timer_getoverrun.3c target=timer_settime.3c
1286 link path=usr/share/man/man3c/timer_gettime.3c target=timer_settime.3c
1287 link path=usr/share/man/man3c/timerclear.3c target=timeradd.3c
1288 link path=usr/share/man/man3c/timercmp.3c target=timeradd.3c
1289 link path=usr/share/man/man3c/timerfd_gettime.3c target=timerfd_create.3c
1290 link path=usr/share/man/man3c/timerfd_settime.3c target=timerfd_create.3c
1291 link path=usr/share/man/man3c/timerisset.3c target=timeradd.3c
1292 link path=usr/share/man/man3c/timersub.3c target=timeradd.3c
1293 link path=usr/share/man/man3c/tmpnam_r.3c target=tmpnam.3c
1294 link path=usr/share/man/man3c/tolower_l.3c target=tolower.3c
1295 link path=usr/share/man/man3c/toupper_l.3c target=toupper.3c
1296 link path=usr/share/man/man3c/towctrans.3c target=wctrans.3c
1297 link path=usr/share/man/man3c/towctrans_l.3c target=wctrans.3c
1298 link path=usr/share/man/man3c/towlower_l.3c target=towlower.3c
1299 link path=usr/share/man/man3c/towupper_l.3c target=towupper.3c
1300 link path=usr/share/man/man3c/ttyname_r.3c target=ttyname.3c
1301 link path=usr/share/man/man3c/twalk.3c target=tsearch.3c
1302 link path=usr/share/man/man3c/tzset.3c target=ctime.3c
1303 link path=usr/share/man/man3c/uconv_u16tou8.3c target=uconv_u16tou32.3c
1304 link path=usr/share/man/man3c/uconv_u32tou16.3c target=uconv_u16tou32.3c
1305 link path=usr/share/man/man3c/uconv_u32tou8.3c target=uconv_u16tou32.3c
1306 link path=usr/share/man/man3c/uconv_u8tou16.3c target=uconv_u16tou32.3c
1307 link path=usr/share/man/man3c/uconv_u8tou32.3c target=uconv_u16tou32.3c
1308 link path=usr/share/man/man3c/ucred_free.3c target=ucred.3c
1309 link path=usr/share/man/man3c/ucred_get.3c target=ucred.3c
1310 link path=usr/share/man/man3c/ucred_getegid.3c target=ucred.3c
1311 link path=usr/share/man/man3c/ucred_geteuid.3c target=ucred.3c
1312 link path=usr/share/man/man3c/ucred_getgroups.3c target=ucred.3c
1313 link path=usr/share/man/man3c/ucred_getlabel.3c target=ucred.3c
1314 link path=usr/share/man/man3c/ucred_getpflags.3c target=ucred.3c
1315 link path=usr/share/man/man3c/ucred_getpid.3c target=ucred.3c
```

```
1316 link path=usr/share/man/man3c/ucred_getprivset.3c target=ucred.3c
1317 link path=usr/share/man/man3c/ucred_getprojid.3c target=ucred.3c
1318 link path=usr/share/man/man3c/ucred_getrgid.3c target=ucred.3c
1319 link path=usr/share/man/man3c/ucred_getruid.3c target=ucred.3c
1320 link path=usr/share/man/man3c/ucred_getsgid.3c target=ucred.3c
1321 link path=usr/share/man/man3c/ucred_getsuid.3c target=ucred.3c
1322 link path=usr/share/man/man3c/ucred_getzoneid.3c target=ucred.3c
1323 link path=usr/share/man/man3c/ucred_size.3c target=ucred.3c
1324 link path=usr/share/man/man3c/ulckpwdf.3c target=lckpwdf.3c
1325 link path=usr/share/man/man3c/ulltostr.3c target=strtol.3c
1326 link path=usr/share/man/man3c/unordered.3c target=isnand.3c
1327 link path=usr/share/man/man3c/updwtmp.3c target=getutxent.3c
1328 link path=usr/share/man/man3c/updwtmpx.3c target=getutxent.3c
1329 link path=usr/share/man/man3c/utmpname.3c target=getutent.3c
1330 link path=usr/share/man/man3c/utmpxname.3c target=getutxent.3c
1331 link path=usr/share/man/man3c/valloc.3c target=malloc.3c
1332 link path=usr/share/man/man3c/vasprintf.3c target=vprintf.3c
1333 link path=usr/share/man/man3c/verr.3c target=err.3c
1334 link path=usr/share/man/man3c/verrx.3c target=err.3c
1335 link path=usr/share/man/man3c/vfprintf.3c target=vprintf.3c
1336 link path=usr/share/man/man3c/vfscanf.3c target=scanf.3c
1337 link path=usr/share/man/man3c/vfwscanf.3c target=fwscanf.3c
1338 link path=usr/share/man/man3c/vscanf.3c target=scanf.3c
1339 link path=usr/share/man/man3c/vsnprintf.3c target=vprintf.3c
1340 link path=usr/share/man/man3c/vsprintf.3c target=vprintf.3c
1341 link path=usr/share/man/man3c/vsscanf.3c target=scanf.3c
1342 link path=usr/share/man/man3c/vswprintf.3c target=vfwprintf.3c
1343 link path=usr/share/man/man3c/vswscanf.3c target=fwscanf.3c
1344 link path=usr/share/man/man3c/vwarn.3c target=err.3c
1345 link path=usr/share/man/man3c/vwarnx.3c target=err.3c
1346 link path=usr/share/man/man3c/vwprintf.3c target=vfwprintf.3c
1347 link path=usr/share/man/man3c/vwscanf.3c target=fwscanf.3c
1348 link path=usr/share/man/man3c/wait4.3c target=wait3.3c
1349 link path=usr/share/man/man3c/warn.3c target=err.3c
1350 link path=usr/share/man/man3c/warnx.3c target=err.3c
1351 link path=usr/share/man/man3c/watof.3c target=wcstod.3c
1352 link path=usr/share/man/man3c/watoi.3c target=wcstol.3c
1353 link path=usr/share/man/man3c/watol.3c target=wcstol.3c
1354 link path=usr/share/man/man3c/watoll.3c target=wcstol.3c
1355 link path=usr/share/man/man3c/wcpncpy.3c target=wcpcpy.3c
1356 link path=usr/share/man/man3c/wcrtomb_l.3c target=wcrtomb.3c
1357 link path=usr/share/man/man3c/wcscasecmp_l.3c target=wcscasecmp.3c
1358 link path=usr/share/man/man3c/wcscat.3c target=wcstring.3c
1359 link path=usr/share/man/man3c/wcschr.3c target=wcstring.3c
1360 link path=usr/share/man/man3c/wcscmp.3c target=wcstring.3c
1361 link path=usr/share/man/man3c/wcscoll_l.3c target=wcscoll.3c
1362 link path=usr/share/man/man3c/wcscpy.3c target=wcstring.3c
1363 link path=usr/share/man/man3c/wcscspn.3c target=wcstring.3c
1364 link path=usr/share/man/man3c/wcsetno.3c target=cset.3c
1365 link path=usr/share/man/man3c/wcsncasecmp.3c target=wcscasecmp.3c
1366 link path=usr/share/man/man3c/wcsncasecmp_l.3c target=wcscasecmp.3c
1367 link path=usr/share/man/man3c/wcsncat.3c target=wcstring.3c
1368 link path=usr/share/man/man3c/wcsncmp.3c target=wcstring.3c
1369 link path=usr/share/man/man3c/wcsncpy.3c target=wcstring.3c
1370 link path=usr/share/man/man3c/wcsnlen.3c target=wcslen.3c
1371 link path=usr/share/man/man3c/wcsnrtombs.3c target=wcsrtombs.3c
1372 link path=usr/share/man/man3c/wcsnrtombs_l.3c target=wcsrtombs.3c
1373 link path=usr/share/man/man3c/wcspbrk.3c target=wcstring.3c
1374 link path=usr/share/man/man3c/wcsrchr.3c target=wcstring.3c
1375 link path=usr/share/man/man3c/wcsrtombs_l.3c target=wcsrtombs.3c
1376 link path=usr/share/man/man3c/wcsspn.3c target=wcstring.3c
1377 link path=usr/share/man/man3c/wcstof.3c target=wcstod.3c
1378 link path=usr/share/man/man3c/wcstok.3c target=wcstring.3c
1379 link path=usr/share/man/man3c/wcstold.3c target=wcstod.3c
1380 link path=usr/share/man/man3c/wcstoll.3c target=wcstol.3c
1381 link path=usr/share/man/man3c/wcstoull.3c target=wcstoul.3c
```



```
1382 link path=usr/share/man/man3c/wcstoumax.3c target=wcstoimax.3c
1383 link path=usr/share/man/man3c/wcswcs.3c target=wcstring.3c
1384 link path=usr/share/man/man3c/wcswidth_l.3c target=wcswidth.3c
1385 link path=usr/share/man/man3c/wctob_l.3c target=wctob.3c
1386 link path=usr/share/man/man3c/wctomb_l.3c target=wctomb.3c
1387 link path=usr/share/man/man3c/wctrans_l.3c target=wctrans.3c
1388 link path=usr/share/man/man3c/wctype_l.3c target=wctype.3c
1389 link path=usr/share/man/man3c/wcwidth_l.3c target=wcwidth.3c
1390 link path=usr/share/man/man3c/windex.3c target=wcstring.3c
1391 link path=usr/share/man/man3c/wordfree.3c target=wordexp.3c
1392 link path=usr/share/man/man3c/wprintf.3c target=fwprintf.3c
1393 link path=usr/share/man/man3c/wrindex.3c target=wcstring.3c
1394 link path=usr/share/man/man3c/wscanf.3c target=fwscanf.3c
1395 link path=usr/share/man/man3c/wscasecmp.3c target=wstring.3c
1396 link path=usr/share/man/man3c/wscat.3c target=wcstring.3c
1397 link path=usr/share/man/man3c/wschr.3c target=wcstring.3c
1398 link path=usr/share/man/man3c/wscmp.3c target=wcstring.3c
1399 link path=usr/share/man/man3c/wscol.3c target=wstring.3c
1400 link path=usr/share/man/man3c/wscoll.3c target=wcscoll.3c
1401 link path=usr/share/man/man3c/wscpy.3c target=wcstring.3c
1402 link path=usr/share/man/man3c/wscspn.3c target=wcstring.3c
1403 link path=usr/share/man/man3c/wsdup.3c target=wstring.3c
1404 link path=usr/share/man/man3c/wslen.3c target=wcstring.3c
1405 link path=usr/share/man/man3c/wsncasecmp.3c target=wstring.3c
1406 link path=usr/share/man/man3c/wsncat.3c target=wcstring.3c
1407 link path=usr/share/man/man3c/wsncmp.3c target=wcstring.3c
1408 link path=usr/share/man/man3c/wsncpy.3c target=wcstring.3c
1409 link path=usr/share/man/man3c/wspbrk.3c target=wcstring.3c
1410 link path=usr/share/man/man3c/wsrchr.3c target=wcstring.3c
1411 link path=usr/share/man/man3c/wsspn.3c target=wcstring.3c
1412 link path=usr/share/man/man3c/wstod.3c target=wcstod.3c
1413 link path=usr/share/man/man3c/wstok.3c target=wcstring.3c
1414 link path=usr/share/man/man3c/wstol.3c target=wcstol.3c
1415 link path=usr/share/man/man3c/wstostr.3c target=strtows.3c
1416 link path=usr/share/man/man3c/wsxfrm.3c target=wcsxfrm.3c
```

```
***********************************************************
   32705 Mon Dec 28 20:02:39 2015
new/usr/src/uts/common/os/cred.c
uts: add a concept of a 'default' set of privileges, separate from 'basic'
***********************************************************
______unchanged_portion_omitted_



 162 /*
 163  * Initialize credentials data structures.
 164  */

 166 void
 167 cred_init(void)
 168 {
 169         priv_init();

 171         crsize = sizeof (cred_t);

 173         if (get_c2audit_load() > 0) {
 174 #ifdef _LP64
 175                 /* assure audit context is 64-bit aligned */
 176                 audoff = (crsize +
 177                     sizeof (int64_t) - 1) & ~(sizeof (int64_t) - 1);
 178 #else   /* _LP64 */
 179                 audoff = crsize;
 180 #endif  /* _LP64 */
 181                 crsize = audoff + sizeof (auditinfo_addr_t);
 182                 crsize = (crsize + sizeof (int) - 1) & ~(sizeof (int) - 1);
 183         }

 185         cred_cache = kmem_cache_create("cred_cache", crsize, 0,
 186             NULL, NULL, NULL, NULL, NULL, 0);

 188         /*
 189          * dummycr is used to copy initial state for creds.
 190          */
 191         dummycr = cralloc();
 192         bzero(dummycr, crsize);
 193         dummycr->cr_ref = 1;
 194         dummycr->cr_uid = (uid_t)-1;
 195         dummycr->cr_gid = (gid_t)-1;
 196         dummycr->cr_ruid = (uid_t)-1;
 197         dummycr->cr_rgid = (gid_t)-1;
 198         dummycr->cr_suid = (uid_t)-1;
 199         dummycr->cr_sgid = (gid_t)-1;


 202         /*
 203          * kcred is used by anything that needs all privileges; it's
 204          * also the template used for crget as it has all the compatible
 205          * sets filled in.
 206          */
 207         kcred = cralloc();

 209         bzero(kcred, crsize);
 210         kcred->cr_ref = 1;

 212         /* kcred is never freed, so we don't need zone_cred_hold here */
 213         kcred->cr_zone = &zone0;

 215         priv_fillset(&CR_LPRIV(kcred));
 216         CR_IPRIV(kcred) = *priv_default; /* XXX: Really needed here? */
 216         CR_IPRIV(kcred) = *priv_basic;
```



```
 218         /* Not a basic privilege, if chown is not restricted add it to I0 */
 219         if (!rstchown)
 220                 priv_addset(&CR_IPRIV(kcred), PRIV_FILE_CHOWN_SELF);

 222         /* Basic privilege, if link is restricted remove it from I0 */
 223         if (rstlink)
 224                 priv_delset(&CR_IPRIV(kcred), PRIV_FILE_LINK_ANY);

 226         CR_EPRIV(kcred) = CR_PPRIV(kcred) = CR_IPRIV(kcred);

 228         CR_FLAGS(kcred) = NET_MAC_AWARE;

 230         /*
 231          * Set up credentials of p0.
 232          */
 233         ttoproc(curthread)->p_cred = kcred;
 234         curthread->t_cred = kcred;

 236         ucredsize = UCRED_SIZE;

 238         mutex_init(&ephemeral_zone_mutex, NULL, MUTEX_DEFAULT, NULL);
 239         zone_key_create(&ephemeral_zone_key, NULL, NULL, destroy_ephemeral_zsd);
 240 }
______unchanged_portion_omitted_
```

```
**********************************************************
   52326 Mon Dec 28 20:02:40 2015
new/usr/src/uts/common/os/exec.c
uts: give privilege macros more sensible names
**********************************************************
______unchanged_portion_omitted_


/*
 * Perform generic exec duties and switchout to object-file specific
 * handler.
 */
int
gexec(
        struct vnode **vpp,
        struct execa *uap,
        struct uarg *args,
        struct intpdata *idatap,
        int level,
        long *execsz,
        caddr_t exec_file,
        struct cred *cred,
        int brand_action)
{
        struct vnode *vp, *execvp = NULL;
        proc_t *pp = ttoproc(curthread);
        struct execsw *eswp;
        int error = 0;
        int suidflags = 0;
        ssize_t resid;
        uid_t uid, gid;
        struct vattr vattr;
        char magbuf[MAGIC_BYTES];
        int setid;
        cred_t *oldcred, *newcred = NULL;
        int privflags = 0;
        int setidfl;
        priv_set_t fset;

        /*
         * If the SNOCD or SUGID flag is set, turn it off and remember the
         * previous setting so we can restore it if we encounter an error.
         */
        if (level == 0 && (pp->p_flag & PSUIDFLAGS)) {
                mutex_enter(&pp->p_lock);
                suidflags = pp->p_flag & PSUIDFLAGS;
                pp->p_flag &= ~PSUIDFLAGS;
                mutex_exit(&pp->p_lock);
        }

        if ((error = execpermissions(*vpp, &vattr, args)) != 0)
                goto bad_noclose;

        /* need to open vnode for stateful file systems */
        if ((error = VOP_OPEN(vpp, FREAD, CRED(), NULL)) != 0)
                goto bad_noclose;
        vp = *vpp;

        /*
         * Note: to support binary compatibility with SunOS a.out
         * executables, we read in the first four bytes, as the
         * magic number is in bytes 2-3.
         */
        if (error = vn_rdwr(UIO_READ, vp, magbuf, sizeof (magbuf),
            (offset_t)0, UIO_SYSSPACE, 0, (rlim64_t)0, CRED(), &resid))
                goto bad;
```



```
        if (resid != 0)
                goto bad;

        if ((eswp = findexec_by_hdr(magbuf)) == NULL)
                goto bad;

        if (level == 0 &&
            (privflags = execsetid(vp, &vattr, &uid, &gid, &fset,
            args->pfcred == NULL ? cred : args->pfcred, args->pathname)) != 0) {

                /* Pfcred is a credential with a ref count of 1 */

                if (args->pfcred != NULL) {
                        privflags |= PRIV_INCREASE|PRIV_RESET;
                        newcred = cred = args->pfcred;
                } else {
                        newcred = cred = crdup(cred);
                }

                /* If we can, drop the PA bit */
                if ((privflags & PRIV_RESET) != 0)
                        priv_adjust_PA(cred);

                if (privflags & PRIV_SETID) {
                        cred->cr_uid = uid;
                        cred->cr_gid = gid;
                        cred->cr_suid = uid;
                        cred->cr_sgid = gid;
                }

                if (privflags & MAC_FLAGS) {
                        if (!(CR_FLAGS(cred) & NET_MAC_AWARE_INHERIT))
                                CR_FLAGS(cred) &= ~NET_MAC_AWARE;
                        CR_FLAGS(cred) &= ~NET_MAC_AWARE_INHERIT;
                }

                /*
                 * Implement the privilege updates:
                 *
                 * Restrict with L:
                 *
                 *      I' = I & L
                 *
                 *      E' = P' = (I' + F) & A
                 *
                 * But if running under ptrace, we cap I and F with P.
                 */
                if ((privflags & (PRIV_RESET|PRIV_FORCED)) != 0) {
                        if ((privflags & PRIV_INCREASE) != 0 &&
                            (pp->p_proc_flag & P_PR_PTRACE) != 0) {
                                priv_intersect(&CR_OPPRIV(cred),
                                    &CR_IPRIV(cred));
                                priv_intersect(&CR_OPPRIV(cred), &fset);
                        }
                        priv_intersect(&CR_LPRIV(cred), &CR_IPRIV(cred));
                        CR_EPRIV(cred) = CR_PPRIV(cred) = CR_IPRIV(cred);
                        if (privflags & PRIV_FORCED) {
                                priv_set_PA(cred);
                                priv_union(&fset, &CR_EPRIV(cred));
                                priv_union(&fset, &CR_PPRIV(cred));
                        }
                        priv_adjust_PA(cred);
                }
        } else if (level == 0 && args->pfcred != NULL) {
                newcred = cred = args->pfcred;
                privflags |= PRIV_INCREASE;
```

```
657                 /* pfcred is not forced to adhere to these settings */
658                 priv_intersect(&CR_LPRIV(cred), &CR_IPRIV(cred));
659                 CR_EPRIV(cred) = CR_PPRIV(cred) = CR_IPRIV(cred);
660                 priv_adjust_PA(cred);
661         }

663         /* SunOS 4.x buy-back */
664         if ((vp->v_vfsp->vfs_flag & VFS_NOSETUID) &&
665             (vattr.va_mode & (VSUID|VSGID))) {
666                 char path[MAXNAMELEN];
667                 refstr_t *mntpt = NULL;
668                 int ret = -1;

670                 bzero(path, sizeof (path));
671                 zone_hold(pp->p_zone);

673                 ret = vnodetopath(pp->p_zone->zone_rootvp, vp, path,
674                     sizeof (path), cred);

676                 /* fallback to mountpoint if a path can't be found */
677                 if ((ret != 0) || (ret == 0 && path[0] == '\0'))
678                         mntpt = vfs_getmntpoint(vp->v_vfsp);

680                 if (mntpt == NULL)
681                         zcmn_err(pp->p_zone->zone_id, CE_NOTE,
682                             "!uid %d: setuid execution not allowed, "
683                             "file=%s", cred->cr_uid, path);
684                 else
685                         zcmn_err(pp->p_zone->zone_id, CE_NOTE,
686                             "!uid %d: setuid execution not allowed, "
687                             "fs=%s, file=%s", cred->cr_uid,
688                             ZONE_PATH_TRANSLATE(refstr_value(mntpt),
689                             pp->p_zone), exec_file);

691                 if (!INGLOBALZONE(pp)) {
692                         /* zone_rootpath always has trailing / */
693                         if (mntpt == NULL)
694                                 cmn_err(CE_NOTE, "!zone: %s, uid: %d "
695                                     "setuid execution not allowed, file=%s%s",
696                                     pp->p_zone->zone_name, cred->cr_uid,
697                                     pp->p_zone->zone_rootpath, path + 1);
698                         else
699                                 cmn_err(CE_NOTE, "!zone: %s, uid: %d "
700                                     "setuid execution not allowed, fs=%s, "
701                                     "file=%s", pp->p_zone->zone_name,
702                                     cred->cr_uid, refstr_value(mntpt),
703                                     exec_file);
704                 }

706                 if (mntpt != NULL)
707                         refstr_rele(mntpt);

709                 zone_rele(pp->p_zone);
710         }

712         /*
713          * execsetid() told us whether or not we had to change the
714          * credentials of the process.  In privflags, it told us
715          * whether we gained any privileges or executed a set-uid executable.
716          */
717         setid = (privflags & (PRIV_SETUGID|PRIV_INCREASE|PRIV_FORCED));

719         /*
720          * Use /etc/system variable to determine if the stack
721          * should be marked as executable by default.
722          */
```



```
723         if (noexec_user_stack)
724                 args->stk_prot &= ~PROT_EXEC;

726         args->execswp = eswp; /* Save execsw pointer in uarg for exec_func */
727         args->ex_vp = vp;

729         /*
730          * Traditionally, the setid flags told the sub processes whether
731          * the file just executed was set-uid or set-gid; this caused
732          * some confusion as the 'setid' flag did not match the SUGID
733          * process flag which is only set when the uids/gids do not match.
734          * A script set-gid/set-uid to the real uid/gid would start with
735          * /dev/fd/X but an executable would happily trust LD_LIBRARY_PATH.
736          * Now we flag those cases where the calling process cannot
737          * be trusted to influence the newly exec'ed process, either
738          * because it runs with more privileges or when the uids/gids
739          * do in fact not match.
740          * This also makes the runtime linker agree with the on exec
741          * values of SNOCD and SUGID.
742          */
743         setidfl = 0;
744         if (cred->cr_uid != cred->cr_ruid || (cred->cr_rgid != cred->cr_gid &&
745             !supgroupmember(cred->cr_gid, cred))) {
746                 setidfl |= EXECSETID_UGIDS;
747         }
748         if (setid & PRIV_SETUGID)
749                 setidfl |= EXECSETID_SETID;
750         if (setid & PRIV_FORCED)
751                 setidfl |= EXECSETID_PRIVS;

753         execvp = pp->p_exec;
754         if (execvp)
755                 VN_HOLD(execvp);

757         error = (*eswp->exec_func)(vp, uap, args, idatap, level, execsz,
758             setidfl, exec_file, cred, brand_action);
759         rw_exit(eswp->exec_lock);
760         if (error != 0) {
761                 if (execvp)
762                         VN_RELE(execvp);
763                 /*
764                  * If this process's p_exec has been set to the vp of
765                  * the executable by exec_func, we will return without
766                  * calling VOP_CLOSE because proc_exit will close it
767                  * on exit.
768                  */
769                 if (pp->p_exec == vp)
770                         goto bad_noclose;
771                 else
772                         goto bad;
773         }

775         if (level == 0) {
776                 uid_t oruid;

778                 if (execvp != NULL) {
779                         /*
780                          * Close the previous executable only if we are
781                          * at level 0.
782                          */
783                         (void) VOP_CLOSE(execvp, FREAD, 1, (offset_t)0,
784                             cred, NULL);
785                 }

787                 mutex_enter(&pp->p_crlock);
```

```
789                 oruid = pp->p_cred->cr_ruid;

791                 if (newcred != NULL) {
792                         /*
793                          * Free the old credentials, and set the new ones.
794                          * Do this for both the process and the (single) thread.
795                          */
796                         crfree(pp->p_cred);
797                         pp->p_cred = cred;      /* cred already held for proc */
798                         crhold(cred);           /* hold new cred for thread */
799                         /*
800                          * DTrace accesses t_cred in probe context.  t_cred
801                          * must always be either NULL, or point to a valid,
802                          * allocated cred structure.
803                          */
804                         oldcred = curthread->t_cred;
805                         curthread->t_cred = cred;
806                         crfree(oldcred);

808                         if (priv_basic_test >= 0 &&
809                             !PRIV_ISMEMBER(&CR_IPRIV(newcred),
809                             !PRIV_ISASSERT(&CR_IPRIV(newcred),
810                             priv_basic_test)) {
811                                 pid_t pid = pp->p_pid;
812                                 char *fn = PTOU(pp)->u_comm;

814                                 cmn_err(CE_WARN, "%s[%d]: exec: basic_test "
815                                     "privilege removed from E/I", fn, pid);
816                         }
817                 }
818                 /*
819                  * On emerging from a successful exec(), the saved
820                  * uid and gid equal the effective uid and gid.
821                  */
822                 cred->cr_suid = cred->cr_uid;
823                 cred->cr_sgid = cred->cr_gid;

825                 /*
826                  * If the real and effective ids do not match, this
827                  * is a setuid process that should not dump core.
828                  * The group comparison is tricky; we prevent the code
829                  * from flagging SNOCD when executing with an effective gid
830                  * which is a supplementary group.
831                  */
832                 if (cred->cr_ruid != cred->cr_uid ||
833                     (cred->cr_rgid != cred->cr_gid &&
834                     !supgroupmember(cred->cr_gid, cred)) ||
835                     (privflags & PRIV_INCREASE) != 0)
836                         suidflags = PSUIDFLAGS;
837                 else
838                         suidflags = 0;

840                 mutex_exit(&pp->p_crlock);
841                 if (newcred != NULL && oruid != newcred->cr_ruid) {
842                         /* Note that the process remains in the same zone. */
843                         mutex_enter(&pidlock);
844                         upcount_dec(oruid, crgetzoneid(newcred));
845                         upcount_inc(newcred->cr_ruid, crgetzoneid(newcred));
846                         mutex_exit(&pidlock);
847                 }
848                 if (suidflags) {
849                         mutex_enter(&pp->p_lock);
850                         pp->p_flag |= suidflags;
851                         mutex_exit(&pp->p_lock);
852                 }
853                 if (setid && (pp->p_proc_flag & P_PR_PTRACE) == 0) {
```



```
854                         /*
855                          * If process is traced via /proc, arrange to
856                          * invalidate the associated /proc vnode.
857                          */
858                         if (pp->p_plist || (pp->p_proc_flag & P_PR_TRACE))
859                                 args->traceinval = 1;
860                 }
861                 if (pp->p_proc_flag & P_PR_PTRACE)
862                         psignal(pp, SIGTRAP);
863                 if (args->traceinval)
864                         prinvalidate(&pp->p_user);
865         }
866         if (execvp)
867                 VN_RELE(execvp);
868         return (0);

870 bad:
871         (void) VOP_CLOSE(vp, FREAD, 1, (offset_t)0, cred, NULL);

873 bad_noclose:
874         if (newcred != NULL)
875                 crfree(newcred);
876         if (error == 0)
877                 error = ENOEXEC;

879         if (suidflags) {
880                 mutex_enter(&pp->p_lock);
881                 pp->p_flag |= suidflags;
882                 mutex_exit(&pp->p_lock);
883         }
884         return (error);
885 }
```

_____unchanged_portion_omitted_

```
**********************************************************
   62951 Mon Dec 28 20:02:40 2015
new/usr/src/uts/common/os/policy.c
uts: give privilege macros more sensible names
**********************************************************
```

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License (the "License").
 * You may not use this file except in compliance with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright (c) 2003, 2010, Oracle and/or its affiliates. All rights reserved.
 * Copyright 2013, Joyent, Inc. All rights reserved.
 */

#include <sys/types.h>
#include <sys/sysmacros.h>
#include <sys/param.h>
#include <sys/systm.h>
#include <sys/cred_impl.h>
#include <sys/vnode.h>
#include <sys/vfs.h>
#include <sys/stat.h>
#include <sys/errno.h>
#include <sys/kmem.h>
#include <sys/user.h>
#include <sys/proc.h>
#include <sys/acct.h>
#include <sys/ipc_impl.h>
#include <sys/cmn_err.h>
#include <sys/debug.h>
#include <sys/policy.h>
#include <sys/kobj.h>
#include <sys/msg.h>
#include <sys/devpolicy.h>
#include <c2/audit.h>
#include <sys/varargs.h>
#include <sys/klpd.h>
#include <sys/modctl.h>
#include <sys/disp.h>
#include <sys/zone.h>
#include <inet/optcom.h>
#include <sys/sdt.h>
#include <sys/vfs.h>
#include <sys/mntent.h>
#include <sys/contract_impl.h>
#include <sys/dld_ioc.h>

/*
 * There are two possible layers of privilege routines and two possible
 * levels of secpolicy.  Plus one other we may not be interested in, so
```



```
 * we may need as many as 6 but no more.
 */
#define MAXPRIVSTACK            6

int priv_debug = 0;
int priv_basic_test = -1;

/*
 * This file contains the majority of the policy routines.
 * Since the policy routines are defined by function and not
 * by privilege, there is quite a bit of duplication of
 * functions.
 *
 * The secpolicy functions must not make assumptions about
 * locks held or not held as any lock can be held while they're
 * being called.
 *
 * Credentials are read-only so no special precautions need to
 * be taken while locking them.
 *
 * When a new policy check needs to be added to the system the
 * following procedure should be followed:
 *
 *              Pick an appropriate secpolicy_*() function
 *                      -> done if one exists.
 *              Create a new secpolicy function, preferably with
 *              a descriptive name using the standard template.
 *              Pick an appropriate privilege for the policy.
 *              If no appropraite privilege exists, define new one
 *              (this should be done with extreme care; in most cases
 *              little is gained by adding another privilege)
 *
 * WHY ROOT IS STILL SPECIAL.
 *
 * In a number of the policy functions, there are still explicit
 * checks for uid 0.  The rationale behind these is that many root
 * owned files/objects hold configuration information which can give full
 * privileges to the user once written to.  To prevent escalation
 * of privilege by allowing just a single privilege to modify root owned
 * objects, we've added these root specific checks where we considered
 * them necessary: modifying root owned files, changing uids to 0, etc.
 *
 * PRIVILEGE ESCALATION AND ZONES.
 *
 * A number of operations potentially allow the caller to achieve
 * privileges beyond the ones normally required to perform the operation.
 * For example, if allowed to create a setuid 0 executable, a process can
 * gain privileges beyond PRIV_FILE_SETID.  Zones, however, place
 * restrictions on the ability to gain privileges beyond those available
 * within the zone through file and process manipulation.  Hence, such
 * operations require that the caller have an effective set that includes
 * all privileges available within the current zone, or all privileges
 * if executing in the global zone.
 *
 * This is indicated in the priv_policy* policy checking functions
 * through a combination of parameters.  The "priv" parameter indicates
 * the privilege that is required, and the "allzone" parameter indicates
 * whether or not all privileges in the zone are required.  In addition,
 * priv can be set to PRIV_ALL to indicate that all privileges are
 * required (regardless of zone).  There are three scenarios of interest:
 * (1) operation requires a specific privilege
 * (2) operation requires a specific privilege, and requires all
 *     privileges available within the zone (or all privileges if in
 *     the global zone)
 * (3) operation requires all privileges, regardless of zone
 *
```

```
 128  * For (1), priv should be set to the specific privilege, and allzone
 129  * should be set to B_FALSE.
 130  * For (2), priv should be set to the specific privilege, and allzone
 131  * should be set to B_TRUE.
 132  * For (3), priv should be set to PRIV_ALL, and allzone should be set
 133  * to B_FALSE.
 134  *
 135  */

 137 /*
 138  * The privileges are checked against the Effective set for
 139  * ordinary processes and checked against the Limit set
 140  * for euid 0 processes that haven't manipulated their privilege
 141  * sets.
 142  */
 143 #define HAS_ALLPRIVS(cr)        priv_isfullset(&CR_OEPRIV(cr))
 144 #define ZONEPRIVS(cr)           ((cr)->cr_zone->zone_privset)
 145 #define HAS_ALLZONEPRIVS(cr)    priv_issubset(ZONEPRIVS(cr), &CR_OEPRIV(cr))
 146 #define HAS_PRIVILEGE(cr, pr)   ((pr) == PRIV_ALL ? \
 147                                         HAS_ALLPRIVS(cr) : \
 148                                         PRIV_ISMEMBER(&CR_OEPRIV(cr), pr))
 148                                         PRIV_ISASSERT(&CR_OEPRIV(cr), pr))

 150 #define FAST_BASIC_CHECK(cr, priv)      \
 151         if (PRIV_ISMEMBER(&CR_OEPRIV(cr), priv)) { \
 151         if (PRIV_ISASSERT(&CR_OEPRIV(cr), priv)) { \
 152                 DTRACE_PROBE2(priv__ok, int, priv, boolean_t, B_FALSE); \
 153                 return (0); \
 154         }

 156 /*
 157  * Policy checking functions.
 158  *
 159  * All of the system's policy should be implemented here.
 160  */

 162 /*
 163  * Private functions which take an additional va_list argument to
 164  * implement an object specific policy override.
 165  */
 166 static int priv_policy_ap(const cred_t *, int, boolean_t, int,
 167     const char *, va_list);
 168 static int priv_policy_va(const cred_t *, int, boolean_t, int,
 169     const char *, ...);

 171 /*
 172  * Generic policy calls
 173  *
 174  * The "bottom" functions of policy control
 175  */
 176 static char *
 177 mprintf(const char *fmt, ...)
 178 {
 179         va_list args;
 180         char *buf;
 181         size_t len;

 183         va_start(args, fmt);
 184         len = vsnprintf(NULL, 0, fmt, args) + 1;
 185         va_end(args);

 187         buf = kmem_alloc(len, KM_NOSLEEP);

 189         if (buf == NULL)
 190                 return (NULL);
```



```
 192         va_start(args, fmt);
 193         (void) vsnprintf(buf, len, fmt, args);
 194         va_end(args);

 196         return (buf);
 197 }
______unchanged_portion_omitted_

 391 /*
 392  * priv_policy_ap()
 393  * return 0 or error.
 394  * See block comment above for a description of "priv" and "allzone" usage.
 395  */
 396 static int
 397 priv_policy_ap(const cred_t *cr, int priv, boolean_t allzone, int err,
 398     const char *msg, va_list ap)
 399 {
 400         if ((HAS_PRIVILEGE(cr, priv) && (!allzone || HAS_ALLZONEPRIVS(cr))) ||
 401             (!servicing_interrupt() &&
 402             priv_policy_override(cr, priv, allzone, ap) == 0)) {
 403                 if ((allzone || priv == PRIV_ALL ||
 404                     !PRIV_ISMEMBER(priv_basic, priv)) &&
 404                     !PRIV_ISASSERT(priv_basic, priv)) &&
 405                     !servicing_interrupt()) {
 406                         PTOU(curproc)->u_acflag |= ASU; /* Needed for SVVS */
 407                         if (AU_AUDITING())
 408                                 audit_priv(priv,
 409                                     allzone ? ZONEPRIVS(cr) : NULL, 1);
 410                 }
 411                 err = 0;
 412                 DTRACE_PROBE2(priv__ok, int, priv, boolean_t, allzone);
 413         } else if (!servicing_interrupt()) {
 414                 /* Failure audited in this procedure */
 415                 priv_policy_err(cr, priv, allzone, msg);
 416         }
 417         return (err);
 418 }
______unchanged_portion_omitted_

 441 /*
 442  * Return B_TRUE for sufficient privileges, B_FALSE for insufficient privileges.
 443  */
 444 boolean_t
 445 priv_policy_choice(const cred_t *cr, int priv, boolean_t allzone)
 446 {
 447         boolean_t res = HAS_PRIVILEGE(cr, priv) &&
 448             (!allzone || HAS_ALLZONEPRIVS(cr));

 450         /* Audit success only */
 451         if (res && AU_AUDITING() &&
 452             (allzone || priv == PRIV_ALL || !PRIV_ISMEMBER(priv_basic, priv)) &&
 452             (allzone || priv == PRIV_ALL || !PRIV_ISASSERT(priv_basic, priv)) &&
 453             !servicing_interrupt()) {
 454                 audit_priv(priv, allzone ? ZONEPRIVS(cr) : NULL, 1);
 455         }
 456         if (res) {
 457                 DTRACE_PROBE2(priv__ok, int, priv, boolean_t, allzone);
 458         } else {
 459                 DTRACE_PROBE2(priv__err, int, priv, boolean_t, allzone);
 460         }
 461         return (res);
 462 }
______unchanged_portion_omitted_

 963 /*
 964  * Like secpolicy_vnode_access() but we get the actual wanted mode and the
```

```
 965  * current mode of the file, not the missing bits.
 966  */
 967 int
 968 secpolicy_vnode_access2(const cred_t *cr, vnode_t *vp, uid_t owner,
 969     mode_t curmode, mode_t wantmode)
 970 {
 971         mode_t mode;

 973         /* Inline the basic privileges tests. */
 974         if ((wantmode & VREAD) &&
 975             !PRIV_ISMEMBER(&CR_OEPRIV(cr), PRIV_FILE_READ) &&
 975             !PRIV_ISASSERT(&CR_OEPRIV(cr), PRIV_FILE_READ) &&
 976             priv_policy_va(cr, PRIV_FILE_READ, B_FALSE, EACCES, NULL,
 977             KLPDARG_VNODE, vp, (char *)NULL, KLPDARG_NOMORE) != 0) {
 978                 return (EACCES);
 979         }

 981         if ((wantmode & VWRITE) &&
 982             !PRIV_ISMEMBER(&CR_OEPRIV(cr), PRIV_FILE_WRITE) &&
 982             !PRIV_ISASSERT(&CR_OEPRIV(cr), PRIV_FILE_WRITE) &&
 983             priv_policy_va(cr, PRIV_FILE_WRITE, B_FALSE, EACCES, NULL,
 984             KLPDARG_VNODE, vp, (char *)NULL, KLPDARG_NOMORE) != 0) {
 985                 return (EACCES);
 986         }

 988         mode = ~curmode & wantmode;

 990         if (mode == 0)
 991                 return (0);

 993         if ((mode & VREAD) && priv_policy_va(cr, PRIV_FILE_DAC_READ, B_FALSE,
 994             EACCES, NULL, KLPDARG_VNODE, vp, (char *)NULL,
 995             KLPDARG_NOMORE) != 0) {
 996                 return (EACCES);
 997         }

 999         if (mode & VWRITE) {
1000                 boolean_t allzone;

1002                 if (owner == 0 && cr->cr_uid != 0)
1003                         allzone = B_TRUE;
1004                 else
1005                         allzone = B_FALSE;
1006                 if (priv_policy_va(cr, PRIV_FILE_DAC_WRITE, allzone, EACCES,
1007                     NULL, KLPDARG_VNODE, vp, (char *)NULL,
1008                     KLPDARG_NOMORE) != 0) {
1009                         return (EACCES);
1010                 }
1011         }

1013         if (mode & VEXEC) {
1014                 /*
1015                  * Directories use file_dac_search to override the execute bit.
1016                  */
1017                 int p = vp->v_type == VDIR ? PRIV_FILE_DAC_SEARCH :
1018                     PRIV_FILE_DAC_EXECUTE;

1020                 return (priv_policy_va(cr, p, B_FALSE, EACCES, NULL,
1021                     KLPDARG_VNODE, vp, (char *)NULL, KLPDARG_NOMORE));
1022         }
1023         return (0);
1024 }
```

unchanged_portion_omitted_

```
**********************************************************
   17857 Mon Dec 28 20:02:41 2015
new/usr/src/uts/common/os/priv.c
uts: add a concept of a 'default' set of privileges, separate from 'basic'
uts: give privilege macros more sensible names
**********************************************************
```

```
   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 (the "License").
   6  * You may not use this file except in compliance with the License.
   7  *
   8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
   9  * or http://www.opensolaris.org/os/licensing.
  10  * See the License for the specific language governing permissions
  11  * and limitations under the License.
  12  *
  13  * When distributing Covered Code, include this CDDL HEADER in each
  14  * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
  15  * If applicable, add the following below this CDDL HEADER, with the
  16  * fields enclosed by brackets "[]" replaced with your own identifying
  17  * information: Portions Copyright [yyyy] [name of copyright owner]
  18  *
  19  * CDDL HEADER END
  20  */
  21 /*
  22  * Copyright 2010 Sun Microsystems, Inc.  All rights reserved.
  23  * Use is subject to license terms.
  24  */

  26 /*
  27  * Privilege implementation.
  28  *
  29  * This file provides the infrastructure for privilege sets and limits
  30  * the number of files that requires to include <sys/cred_impl.h> and/or
  31  * <sys/priv_impl.h>.
  32  *
  33  * The Solaris privilege mechanism has been designed in a
  34  * future proof manner.  While the kernel may use fixed size arrays
  35  * and fixed bitmasks and bit values, the representation of those
  36  * is kernel private.  All external interfaces as well as K-to-K interfaces
  37  * have been constructed in a manner to provide the maximum flexibility.
  38  *
  39  * There can be X privilege sets each containing Y 32 bit words.
  40  * <X, Y> are constant for a kernel invocation.
  41  *
  42  * As a consequence, all privilege set manipulation happens in functions
  43  * below.
  44  *
  45  */

  47 #include <sys/systm.h>
  48 #include <sys/ddi.h>
  49 #include <sys/kmem.h>
  50 #include <sys/sunddi.h>
  51 #include <sys/errno.h>
  52 #include <sys/debug.h>
  53 #include <sys/priv_impl.h>
  54 #include <sys/procfs.h>
  55 #include <sys/policy.h>
  56 #include <sys/cred_impl.h>
  57 #include <sys/devpolicy.h>
  58 #include <sys/atomic.h>

  60 /*
```



```
  61  * Privilege name to number mapping table consists in the generated
  62  * priv_const.c file.  This lock protects against updates of the privilege
  63  * names and counts; all other priv_info fields are read-only.
  64  * The actual protected values are:
  65  *      global variable nprivs
  66  *      the priv_max field
  67  *      the priv_names field
  68  *      the priv names info item (cnt/strings)
  69  */
  70 krwlock_t privinfo_lock;

  72 static boolean_t priv_valid(const cred_t *);

  74 priv_set_t priv_fullset;        /* set of all privileges */
  75 priv_set_t priv_unsafe; /* unsafe to exec set-uid root if these are not in L */

  77 /*
  78  * Privilege initialization functions.
  79  * Called from common/os/cred.c when cred_init is called.
  80  */

  82 void
  83 priv_init(void)
  84 {
  85 #ifdef DEBUG
  86         int alloc_test_priv = 1;
  87 #else
  88         int alloc_test_priv = priv_debug;
  89 #endif
  90         rw_init(&privinfo_lock, NULL, RW_DRIVER, NULL);

  92         PRIV_BASIC_ADDSET(priv_basic);

  94         /*
  95          * The "default" set is the basic privileges + any 'default'
  96          * privileges.  with no traditional unix connotations.
  97          */
  98         PRIV_BASIC_ADDSET(priv_default);
  99         PRIV_DEFAULT_ADDSET(priv_default);

 101         PRIV_UNSAFE_ADDSET(&priv_unsafe);
  92         PRIV_BASIC_ASSERT(priv_basic);
  93         PRIV_UNSAFE_ASSERT(&priv_unsafe);
 102         priv_fillset(&priv_fullset);

 104         /*
 105          * When booting with priv_debug set or in a DEBUG kernel, then we'll
 106          * add an additional basic privilege and we verify that it is always
 107          * present in E.
 108          */
 109         if (alloc_test_priv != 0 &&
 110             (priv_basic_test = priv_getbyname("basic_test", PRIV_ALLOC)) >= 0) {
 111                 priv_addset(priv_basic, priv_basic_test);
 112                 priv_addset(priv_default, priv_basic_test);
 113 #endif /* ! codereview */
 114         }

 116         devpolicy_init();
 117 }

 119 /* Utility functions: privilege sets as opaque data types */

 121 /*
 122  * Guts of prgetprivsize.
 123  */
 124 int
```

```
125 priv_prgetprivsize(prpriv_t *tmpl)
126 {
127         return (sizeof (prpriv_t) +
128             PRIV_SETBYTES - sizeof (priv_chunk_t) +
129             (tmpl ? tmpl->pr_infosize : priv_info->priv_infosize));
130 }

132 /*
133  * Guts of prgetpriv.
134  */
135 void
136 cred2prpriv(const cred_t *cp, prpriv_t *pr)
137 {
138         priv_set_t *psa;
139         int i;

141         pr->pr_nsets = PRIV_NSET;
142         pr->pr_setsize = PRIV_SETSIZE;
143         pr->pr_infosize = priv_info->priv_infosize;

145         psa = (priv_set_t *)pr->pr_sets;

147         for (i = 0; i < PRIV_NSET; i++)
148                 psa[i] = *priv_getset(cp, i);

150         priv_getinfo(cp, (char *)pr + PRIV_PRPRIV_INFO_OFFSET(pr));
151 }

153 /*
154  * Guts of pr_spriv:
155  *
156  * Set the privileges of a process.
157  *
158  * In order to set the privileges, the setting process will need to
159  * have those privileges in its effective set in order to prevent
160  * specially privileged processes to easily gain additional privileges.
161  * Pre-existing privileges can be retained.  To change any privileges,
162  * PRIV_PROC_OWNER needs to be asserted.
163  *
164  * In formula:
165  *
166  *      S' <= S || S' <= S + Ea
167  *
168  * the new set must either be subset of the old set or a subset of
169  * the oldset merged with the effective set of the acting process; or just:
170  *
171  *      S' <= S + Ea
172  *
173  * It's not legal to grow the limit set this way.
174  *
175  */
176 int
177 priv_pr_spriv(proc_t *p, prpriv_t *prpriv, const cred_t *cr)
178 {
179         cred_t *oldcred;
180         cred_t *newcred;
181         int i;
182         int err = EPERM;
183         cred_priv_t *cp, *ocp;
184         priv_set_t eset;

186         ASSERT(MUTEX_HELD(&p->p_lock));

188         /*
189          * Set must have proper dimension; infosize must be absent
190          * or properly sized.
```



```
191          */
192         if (prpriv->pr_nsets != PRIV_NSET ||
193             prpriv->pr_setsize != PRIV_SETSIZE ||
194             (prpriv->pr_infosize & (sizeof (uint32_t) - 1)) != 0 ||
195             prpriv->pr_infosize > priv_info->priv_infosize ||
196             prpriv->pr_infosize < 0)
197                 return (EINVAL);

199         mutex_exit(&p->p_lock);

201         if (priv_proc_cred_perm(cr, p, &oldcred, VWRITE) != 0) {
202                 mutex_enter(&p->p_lock);
203                 return (EPERM);
204         }

206         newcred = crdup(oldcred);

208         /* Copy the privilege sets from prpriv to newcred */
209         bcopy(prpriv->pr_sets, CR_PRIVSETS(newcred), PRIV_SETBYTES);

211         cp = &newcred->cr_priv;
212         ocp = &oldcred->cr_priv;
213         eset = CR_OEPRIV(cr);

215         priv_intersect(&CR_LPRIV(oldcred), &eset);

217         /*
218          * Verify the constraints laid out:
219          * for the limit set, we require that the new set is a subset
220          * of the old limit set.
221          * for all other sets, we require that the new set is either a
222          * subset of the old set or a subset of the intersection of
223          * the old limit set and the effective set of the acting process.
224          */
225         for (i = 0; i < PRIV_NSET; i++)
226                 if (!priv_issubset(&cp->crprivs[i], &ocp->crprivs[i]) &&
227                     (i == PRIV_LIMIT || !priv_issubset(&cp->crprivs[i], &eset)))
228                         break;

230         crfree(oldcred);

232         if (i < PRIV_NSET || !priv_valid(newcred))
233                 goto err;

235         /* Load the settable privilege information */
236         if (prpriv->pr_infosize > 0) {
237                 char *x = (char *)prpriv + PRIV_PRPRIV_INFO_OFFSET(prpriv);
238                 char *lastx = x + prpriv->pr_infosize;

240                 while (x < lastx) {
241                         priv_info_t *pi = (priv_info_t *)x;
242                         priv_info_uint_t *pii;

244                         switch (pi->priv_info_type) {
245                         case PRIV_INFO_FLAGS:
246                                 pii = (priv_info_uint_t *)x;
247                                 if (pii->info.priv_info_size != sizeof (*pii)) {
248                                         err = EINVAL;
249                                         goto err;
250                                 }
251                                 CR_FLAGS(newcred) &= ~PRIV_USER;
252                                 CR_FLAGS(newcred) |= (pii->val & PRIV_USER);
253                                 break;
254                         default:
255                                 err = EINVAL;
256                                 goto err;
```

```
257                         }
258                         /* Guarantee alignment and forward progress */
259                         if ((pi->priv_info_size & (sizeof (uint32_t) - 1)) ||
260                             pi->priv_info_size < sizeof (*pi) ||
261                             lastx - x > pi->priv_info_size) {
262                                 err = EINVAL;
263                                 goto err;
264                         }
266                         x += pi->priv_info_size;
267                 }
268         }
270         /*
271          * We'll try to copy the privilege aware flag; but since the
272          * privileges sets are all individually set, they are set
273          * as if we're privilege aware.  If PRIV_AWARE wasn't set
274          * or was explicitely unset, we need to set the flag and then
275          * try to get rid of it.
276          */
277         if ((CR_FLAGS(newcred) & PRIV_AWARE) == 0) {
278                 CR_FLAGS(newcred) |= PRIV_AWARE;
279                 priv_adjust_PA(newcred);
280         }
282         mutex_enter(&p->p_crlock);
283         oldcred = p->p_cred;
284         p->p_cred = newcred;
285         mutex_exit(&p->p_crlock);
286         crfree(oldcred);
288         mutex_enter(&p->p_lock);
289         return (0);
291 err:
292         crfree(newcred);
293         mutex_enter(&p->p_lock);
294         return (err);
295 }
297 priv_impl_info_t
298 *priv_hold_implinfo(void)
299 {
300         rw_enter(&privinfo_lock, RW_READER);
301         return (priv_info);
302 }
304 void
305 priv_release_implinfo(void)
306 {
307         rw_exit(&privinfo_lock);
308 }
310 size_t
311 priv_get_implinfo_size(void)
312 {
313         return (privinfosize);
314 }
317 /*
318  * Return the nth privilege set
319  */
320 const priv_set_t *
321 priv_getset(const cred_t *cr, int set)
322 {
```



```
323         ASSERT(PRIV_VALIDSET(set));
325         if ((CR_FLAGS(cr) & PRIV_AWARE) == 0)
326                 switch (set) {
327                 case PRIV_EFFECTIVE:
328                         return (&CR_OEPRIV(cr));
329                 case PRIV_PERMITTED:
330                         return (&CR_OPPRIV(cr));
331                 }
332         return (&CR_PRIVS(cr)->crprivs[set]);
333 }
335 /*
336  * Buf must be allocated by caller and contain sufficient space to
337  * contain all additional info structures using priv_info.priv_infosize.
338  * The buffer must be properly aligned.
339  */
340 /*ARGSUSED*/
341 void
342 priv_getinfo(const cred_t *cr, void *buf)
343 {
344         struct priv_info_uint *ii;
346         ii = buf;
347         ii->val = CR_FLAGS(cr);
348         ii->info.priv_info_size = (uint32_t)sizeof (*ii);
349         ii->info.priv_info_type = PRIV_INFO_FLAGS;
350 }
352 int
353 priv_getbyname(const char *name, uint_t flag)
354 {
355         int i;
356         int wheld = 0;
357         int len;
358         char *p;
360         if (flag != 0 && flag != PRIV_ALLOC)
361                 return (-EINVAL);
363         if (strncasecmp(name, "priv_", 5) == 0)
364                 name += 5;
366         rw_enter(&privinfo_lock, RW_READER);
367 rescan:
368         for (i = 0; i < nprivs; i++)
369                 if (strcasecmp(priv_names[i], name) == 0) {
370                         rw_exit(&privinfo_lock);
371                         return (i);
372                 }
375         if (!wheld) {
376                 if (!(flag & PRIV_ALLOC)) {
377                         rw_exit(&privinfo_lock);
378                         return (-EINVAL);
379                 }
381                 /* check length, validity and available space */
382                 len = strlen(name) + 1;
384                 if (len > PRIVNAME_MAX) {
385                         rw_exit(&privinfo_lock);
386                         return (-ENAMETOOLONG);
387                 }
```

```
389                 for (p = (char *)name; *p != '\0'; p++) {
390                         char c = *p;

392                         if (!((c >= 'A' && c <= 'Z') ||
393                             (c >= 'a' && c <= 'z') ||
394                             (c >= '0' && c <= '9') ||
395                             c == '_')) {
396                                 rw_exit(&privinfo_lock);
397                                 return (-EINVAL);
398                         }
399                 }

401                 if (!rw_tryupgrade(&privinfo_lock)) {
402                         rw_exit(&privinfo_lock);
403                         rw_enter(&privinfo_lock, RW_WRITER);
404                         wheld = 1;
405                         /* Someone may have added our privilege */
406                         goto rescan;
407                 }
408         }

410         if (nprivs == MAX_PRIVILEGE || len + privbytes > maxprivbytes) {
411                 rw_exit(&privinfo_lock);
412                 return (-ENOMEM);
413         }

415         priv_names[i] = p = priv_str + privbytes;

417         bcopy(name, p, len);

419         /* make the priv_names[i] and privilege name globally visible */
420         membar_producer();

422         /* adjust priv count and bytes count */
423         priv_ninfo->cnt = priv_info->priv_max = ++nprivs;
424         privbytes += len;

426         rw_exit(&privinfo_lock);
427         return (i);
428 }

430 /*
431  * We can't afford locking the privileges here because of the locations
432  * we call this from; so we make sure that the privileges table
433  * is visible to us; it is made visible before the value of nprivs is
434  * updated.
435  */
436 const char *
437 priv_getbynum(int priv)
438 {
439         int maxpriv = nprivs;

441         membar_consumer();

443         if (priv >= 0 && priv < maxpriv)
444                 return (priv_names[priv]);

446         return (NULL);
447 }

449 const char *
450 priv_getsetbynum(int setno)
451 {
452         if (!PRIV_VALIDSET(setno))
453                 return (NULL);
```



```
455         return (priv_setnames[setno]);
456 }

458 /*
459  * Privilege sanity checking when setting: E <= P.
460  */
461 static boolean_t
462 priv_valid(const cred_t *cr)
463 {
464         return (priv_issubset(&CR_EPRIV(cr), &CR_PPRIV(cr)));
465 }

467 /*
468  * Privilege manipulation functions
469  *
470  * Without knowing the details of the privilege set implementation,
471  * opaque pointers can be used to manipulate sets at will.
472  */
473 void
474 priv_emptyset(priv_set_t *set)
475 {
476         bzero(set, sizeof (*set));
477 }

479 void
480 priv_fillset(priv_set_t *set)
481 {
482         int i;

484         /* memset? */
485         for (i = 0; i < PRIV_SETSIZE; i++)
486                 set->pbits[i] = ~(priv_chunk_t)0;
487 }

489 void
490 priv_addset(priv_set_t *set, int priv)
491 {
492         ASSERT(priv >= 0 && priv < MAX_PRIVILEGE);
493         __PRIV_ADDSET(set, priv);
104         __PRIV_ASSERT(set, priv);
494 }

496 void
497 priv_delset(priv_set_t *set, int priv)
498 {
499         ASSERT(priv >= 0 && priv < MAX_PRIVILEGE);
500         __PRIV_DELSET(set, priv);
111         __PRIV_CLEAR(set, priv);
501 }

503 boolean_t
504 priv_ismember(const priv_set_t *set, int priv)
505 {
506         ASSERT(priv >= 0 && priv < MAX_PRIVILEGE);
507         return (__PRIV_ISMEMBER(set, priv) ? B_TRUE : B_FALSE);
118         return (__PRIV_ISASSERT(set, priv) ? B_TRUE : B_FALSE);
508 }
______unchanged_portion_omitted_
```

```
**********************************************************
   20840 Mon Dec 28 20:02:42 2015
new/usr/src/uts/common/os/priv_defs
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
```

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License (the "License").
 * You may not use this file except in compliance with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright (c) 2003, 2010, Oracle and/or its affiliates. All rights reserved.
 * Copyright 2015, Joyent, Inc. All rights reserved.
 *
INSERT COMMENT
 */

#
# Privileges can be added to this file at any location, not
# necessarily at the end.  For patches, it is probably best to
# add the new privilege at the end; for ordinary releases privileges
# should be ordered alphabetically.
#

privilege PRIV_CONTRACT_EVENT

        Allows a process to request critical events without limitation.
        Allows a process to request reliable delivery of all events on
        any event queue.

privilege PRIV_CONTRACT_IDENTITY

        Allows a process to set the service FMRI value of a process
        contract template.

privilege PRIV_CONTRACT_OBSERVER

        Allows a process to observe contract events generated by
        contracts created and owned by users other than the process's
        effective user ID.
        Allows a process to open contract event endpoints belonging to
        contracts created and owned by users other than the process's
        effective user ID.

privilege PRIV_CPC_CPU

        Allow a process to access per-CPU hardware performance counters.

privilege PRIV_DTRACE_KERNEL

        Allows DTrace kernel-level tracing.
```



```
privilege PRIV_DTRACE_PROC

        Allows DTrace process-level tracing.
        Allows process-level tracing probes to be placed and enabled in
        processes to which the user has permissions.

privilege PRIV_DTRACE_USER

        Allows DTrace user-level tracing.
        Allows use of the syscall and profile DTrace providers to
        examine processes to which the user has permissions.

privilege PRIV_FILE_CHOWN

        Allows a process to change a file's owner user ID.
        Allows a process to change a file's group ID to one other than
        the process' effective group ID or one of the process'
        supplemental group IDs.

privilege PRIV_FILE_CHOWN_SELF

        Allows a process to give away its files; a process with this
        privilege will run as if {_POSIX_CHOWN_RESTRICTED} is not
        in effect.

privilege PRIV_FILE_DAC_EXECUTE

        Allows a process to execute an executable file whose permission
        bits or ACL do not allow the process execute permission.

privilege PRIV_FILE_DAC_READ

        Allows a process to read a file or directory whose permission
        bits or ACL do not allow the process read permission.

privilege PRIV_FILE_DAC_SEARCH

        Allows a process to search a directory whose permission bits or
        ACL do not allow the process search permission.

privilege PRIV_FILE_DAC_WRITE

        Allows a process to write a file or directory whose permission
        bits or ACL do not allow the process write permission.
        In order to write files owned by uid 0 in the absence of an
        effective uid of 0 ALL privileges are required.

privilege PRIV_FILE_DOWNGRADE_SL

        Allows a process to set the sensitivity label of a file or
        directory to a sensitivity label that does not dominate the
        existing sensitivity label.
        This privilege is interpreted only if the system is configured
        with Trusted Extensions.

privilege PRIV_FILE_FLAG_SET

        Allows a process to set immutable, nounlink or appendonly
        file attributes.

basic privilege PRIV_FILE_LINK_ANY

        Allows a process to create hardlinks to files owned by a uid
        different from the process' effective uid.
```

```
128 privilege PRIV_FILE_OWNER

130         Allows a process which is not the owner of a file or directory
131         to perform the following operations that are normally permitted
132         only for the file owner: modify that file's access and
133         modification times; remove or rename a file or directory whose
134         parent directory has the ''save text image after execution''
135         (sticky) bit set; mount a ''namefs'' upon a file; modify
136         permission bits or ACL except for the set-uid and set-gid
137         bits.

139 basic privilege PRIV_FILE_READ

141         Allows a process to read objects in the filesystem.

143 privilege PRIV_FILE_SETID

145         Allows a process to change the ownership of a file or write to
146         a file without the set-user-ID and set-group-ID bits being
147         cleared.
148         Allows a process to set the set-group-ID bit on a file or
149         directory whose group is not the process' effective group or
150         one of the process' supplemental groups.
151         Allows a process to set the set-user-ID bit on a file with
152         different ownership in the presence of PRIV_FILE_OWNER.
153         Additional restrictions apply when creating or modifying a
154         set-uid 0 file.

156 privilege PRIV_FILE_UPGRADE_SL

158         Allows a process to set the sensitivity label of a file or
159         directory to a sensitivity label that dominates the existing
160         sensitivity label.
161         This privilege is interpreted only if the system is configured
162         with Trusted Extensions.

164 basic privilege PRIV_FILE_WRITE

166         Allows a process to modify objects in the filesystem.

168 privilege PRIV_GRAPHICS_ACCESS

170         Allows a process to make privileged ioctls to graphics devices.
171         Typically only xserver process needs to have this privilege.
172         A process with this privilege is also allowed to perform
173         privileged graphics device mappings.

175 privilege PRIV_GRAPHICS_MAP

177         Allows a process to perform privileged mappings through a
178         graphics device.

180 privilege PRIV_IPC_DAC_READ

182         Allows a process to read a System V IPC
183         Message Queue, Semaphore Set, or Shared Memory Segment whose
184         permission bits do not allow the process read permission.
185         Allows a process to read remote shared memory whose
186         permission bits do not allow the process read permission.

188 privilege PRIV_IPC_DAC_WRITE

190         Allows a process to write a System V IPC
191         Message Queue, Semaphore Set, or Shared Memory Segment whose
192         permission bits do not allow the process write permission.
193         Allows a process to read remote shared memory whose
```



```
194         permission bits do not allow the process write permission.
195         Additional restrictions apply if the owner of the object has uid 0
196         and the effective uid of the current process is not 0.

198 privilege PRIV_IPC_OWNER

200         Allows a process which is not the owner of a System
201         V IPC Message Queue, Semaphore Set, or Shared Memory Segment to
202         remove, change ownership of, or change permission bits of the
203         Message Queue, Semaphore Set, or Shared Memory Segment.
204         Additional restrictions apply if the owner of the object has uid 0
205         and the effective uid of the current process is not 0.

207 basic privilege PRIV_NET_ACCESS

209         Allows a process to open a TCP, UDP, SDP or SCTP network endpoint.

211 privilege PRIV_NET_BINDMLP

213         Allow a process to bind to a port that is configured as a
214         multi-level port(MLP) for the process's zone. This privilege
215         applies to both shared address and zone-specific address MLPs.
216         See tnzonecfg(4) from the Trusted Extensions manual pages for
217         information on configuring MLP ports.
218         This privilege is interpreted only if the system is configured
219         with Trusted Extensions.

221 privilege PRIV_NET_ICMPACCESS

223         Allows a process to send and receive ICMP packets.

225 privilege PRIV_NET_MAC_AWARE

227         Allows a process to set NET_MAC_AWARE process flag by using
228         setpflags(2). This privilege also allows a process to set
229         SO_MAC_EXEMPT socket option by using setsockopt(3SOCKET).
230         The NET_MAC_AWARE process flag and the SO_MAC_EXEMPT socket
231         option both allow a local process to communicate with an
232         unlabeled peer if the local process' label dominates the
233         peer's default label, or if the local process runs in the
234         global zone.
235         This privilege is interpreted only if the system is configured
236         with Trusted Extensions.

238 privilege PRIV_NET_MAC_IMPLICIT

240         Allows a process to set SO_MAC_IMPLICIT option by using
241         setsockopt(3SOCKET).  This allows a privileged process to
242         transmit implicitly-labeled packets to a peer.
243         This privilege is interpreted only if the system is configured
244         with Trusted Extensions.

246 privilege PRIV_NET_OBSERVABILITY

248         Allows a process to access /dev/lo0 and the devices in /dev/ipnet/
249         while not requiring them to need PRIV_NET_RAWACCESS.

251 privilege PRIV_NET_PRIVADDR

253         Allows a process to bind to a privileged port
254         number. The privilege port numbers are 1-1023 (the traditional
255         UNIX privileged ports) as well as those ports marked as
256         "udp/tcp_extra_priv_ports" with the exception of the ports
257         reserved for use by NFS.

259 privilege PRIV_NET_RAWACCESS
```

```
261         Allows a process to have direct access to the network layer.

263 unsafe privilege PRIV_PROC_AUDIT

265         Allows a process to generate audit records.
266         Allows a process to get its own audit pre-selection information.

268 privilege PRIV_PROC_CHROOT

270         Allows a process to change its root directory.

272 privilege PRIV_PROC_CLOCK_HIGHRES

274         Allows a process to use high resolution timers.

276 basic privilege PRIV_PROC_EXEC

278         Allows a process to call execve().

280 basic privilege PRIV_PROC_FORK

282         Allows a process to call fork1()/forkall()/vfork()

284 basic privilege PRIV_PROC_INFO

286         Allows a process to examine the status of processes other
287         than those it can send signals to.  Processes which cannot
288         be examined cannot be seen in /proc and appear not to exist.

290 privilege PRIV_PROC_LOCK_MEMORY

292         Allows a process to lock pages in physical memory.

294 privilege PRIV_PROC_MEMINFO

296         Allows a process to access physical memory information.

298 privilege PRIV_PROC_OWNER

300         Allows a process to send signals to other processes, inspect
301         and modify process state to other processes regardless of
302         ownership.  When modifying another process, additional
303         restrictions apply:  the effective privilege set of the
304         attaching process must be a superset of the target process'
305         effective, permitted and inheritable sets; the limit set must
306         be a superset of the target's limit set; if the target process
307         has any uid set to 0 all privilege must be asserted unless the
308         effective uid is 0.
309         Allows a process to bind arbitrary processes to CPUs.

311 # XXX: This is made default merely for test purposes.  DO NOT LEAVE HERE
312 default privilege PRIV_PROC_PRIOUP
311 privilege PRIV_PROC_PRIOUP

314         Allows a process to elevate its priority above its current level.

316 privilege PRIV_PROC_PRIOCNTL

318         Allows all that PRIV_PROC_PRIOUP allows.
319         Allows a process to change its scheduling class to any scheduling class,
320         including the RT class.

322 basic privilege PRIV_PROC_SESSION

324         Allows a process to send signals or trace processes outside its
```



```
325         session.

327 unsafe privilege PRIV_PROC_SETID

329         Allows a process to set its uids at will.
330         Assuming uid 0 requires all privileges to be asserted.

332 privilege PRIV_PROC_TASKID

334         Allows a process to assign a new task ID to the calling process.

336 privilege PRIV_PROC_ZONE

338         Allows a process to trace or send signals to processes in
339         other zones.

341 privilege PRIV_SYS_ACCT

343         Allows a process to enable and disable and manage accounting through
344         acct(2), getacct(2), putacct(2) and wracct(2).

346 privilege PRIV_SYS_ADMIN

348         Allows a process to perform system administration tasks such
349         as setting node and domain name and specifying nscd and coreadm
350         settings.

352 privilege PRIV_SYS_AUDIT

354         Allows a process to start the (kernel) audit daemon.
355         Allows a process to view and set audit state (audit user ID,
356         audit terminal ID, audit sessions ID, audit pre-selection mask).
357         Allows a process to turn off and on auditing.
358         Allows a process to configure the audit parameters (cache and
359         queue sizes, event to class mappings, policy options).

361 privilege PRIV_SYS_CONFIG

363         Allows a process to perform various system configuration tasks.
364         Allows a process to add and remove swap devices; when adding a swap
365         device, a process must also have sufficient privileges to read from
366         and write to the swap device.

368 privilege PRIV_SYS_DEVICES

370         Allows a process to successfully call a kernel module that
371         calls the kernel drv_priv(9F) function to check for allowed
372         access.
373         Allows a process to open the real console device directly.
374         Allows a process to open devices that have been exclusively opened.

376 privilege PRIV_SYS_IPC_CONFIG

378         Allows a process to increase the size of a System V IPC Message
379         Queue buffer.

381 privilege PRIV_SYS_LINKDIR

383         Allows a process to unlink and link directories.

385 privilege PRIV_SYS_MOUNT

387         Allows filesystem specific administrative procedures, such as
388         filesystem configuration ioctls, quota calls and creation/deletion
389         of snapshots.
390         Allows a process to mount and unmount filesystems which would
```

```
391         otherwise be restricted (i.e., most filesystems except
392         namefs).
393         A process performing a mount operation needs to have
394         appropriate access to the device being mounted (read-write for
395         "rw" mounts, read for "ro" mounts).
396         A process performing any of the aforementioned
397         filesystem operations needs to have read/write/owner
398         access to the mount point.
399         Only regular files and directories can serve as mount points
400         for processes which do not have all zone privileges asserted.
401         Unless a process has all zone privileges, the mount(2)
402         system call will force the "nosuid" and "restrict" options, the
403         latter only for autofs mountpoints.
404         Regardless of privileges, a process running in a non-global zone may
405         only control mounts performed from within said zone.
406         Outside the global zone, the "nodevices" option is always forced.

408 privilege PRIV_SYS_IPTUN_CONFIG

410         Allows a process to configure IP tunnel links.

412 privilege PRIV_SYS_DL_CONFIG

414         Allows a process to configure all classes of datalinks, including
415         configuration allowed by PRIV_SYS_IPTUN_CONFIG.

417 privilege PRIV_SYS_IP_CONFIG

419         Allows a process to configure a system's IP interfaces and routes.
420         Allows a process to configure network parameters using ndd.
421         Allows a process access to otherwise restricted information using ndd.
422         Allows a process to configure IPsec.
423         Allows a process to pop anchored STREAMs modules with matching zoneid.

425 privilege PRIV_SYS_NET_CONFIG

427         Allows all that PRIV_SYS_IP_CONFIG, PRIV_SYS_DL_CONFIG, and
428         PRIV_SYS_PPP_CONFIG allow.
429         Allows a process to push the rpcmod STREAMs module.
430         Allows a process to INSERT/REMOVE STREAMs modules on locations other
431         than the top of the module stack.

433 privilege PRIV_SYS_NFS

435         Allows a process to perform Sun private NFS specific system calls.
436         Allows a process to bind to ports reserved by NFS: ports 2049 (nfs)
437         and port 4045 (lockd).

439 privilege PRIV_SYS_PPP_CONFIG

441         Allows a process to create and destroy PPP (sppp) interfaces.
442         Allows a process to configure PPP tunnels (sppptun).

444 privilege PRIV_SYS_RES_BIND

446         Allows a process to bind processes to processor sets.

448 privilege PRIV_SYS_RES_CONFIG

450         Allows all that PRIV_SYS_RES_BIND allows.
451         Allows a process to create and delete processor sets, assign
452         CPUs to processor sets and override the PSET_NOESCAPE property.
453         Allows a process to change the operational status of CPUs in
454         the system using p_online(2).
455         Allows a process to configure resource pools and to bind
456         processes to pools
```



```
458 unsafe privilege PRIV_SYS_RESOURCE

460         Allows a process to modify the resource limits specified
461         by setrlimit(2) and setrctl(2) without restriction.
462         Allows a process to exceed the per-user maximum number of
463         processes.
464         Allows a process to extend or create files on a filesystem that
465         has less than minfree space in reserve.

467 privilege PRIV_SYS_SMB

469         Allows a process to access the Sun private SMB kernel module.
470         Allows a process to bind to ports reserved by NetBIOS and SMB:
471         ports 137 (NBNS), 138 (NetBIOS Datagram Service), 139 (NetBIOS
472         Session Service and SMB-over-NBT) and 445 (SMB-over-TCP).

474 privilege PRIV_SYS_SUSER_COMPAT

476         Allows a process to successfully call a third party loadable module
477         that calls the kernel suser() function to check for allowed access.
478         This privilege exists only for third party loadable module
479         compatibility and is not used by Solaris proper.

481 privilege PRIV_SYS_TIME

483         Allows a process to manipulate system time using any of the
484         appropriate system calls: stime, adjtime, ntp_adjtime and
485         the IA specific RTC calls.

487 privilege PRIV_SYS_TRANS_LABEL

489         Allows a process to translate labels that are not dominated
490         by the process' sensitivity label to and from an external
491         string form.
492         This privilege is interpreted only if the system is configured
493         with Trusted Extensions.

495 privilege PRIV_VIRT_MANAGE

497         Allows a process to manage virtualized environments such as
498         xVM(5).

500 privilege PRIV_WIN_COLORMAP

502         Allows a process to override colormap restrictions.
503         Allows a process to install or remove colormaps.
504         Allows a process to retrieve colormap cell entries allocated
505         by other processes.
506         This privilege is interpreted only if the system is configured
507         with Trusted Extensions.

509 privilege PRIV_WIN_CONFIG

511         Allows a process to configure or destroy resources that are
512         permanently retained by the X server.
513         Allows a process to use SetScreenSaver to set the screen
514         saver timeout value.
515         Allows a process to use ChangeHosts to modify the display
516         access control list.
517         Allows a process to use GrabServer.
518         Allows a process to use the SetCloseDownMode request which
519         may retain window, pixmap, colormap, property, cursor, font,
520         or graphic context resources.
521         This privilege is interpreted only if the system is configured
522         with Trusted Extensions.
```

```
524 privilege PRIV_WIN_DAC_READ

526         Allows a process to read from a window resource that it does
527         not own (has a different user ID).
528         This privilege is interpreted only if the system is configured
529         with Trusted Extensions.

531 privilege PRIV_WIN_DAC_WRITE

533         Allows a process to write to or create a window resource that
534         it does not own (has a different user ID). A newly created
535         window property is created with the window's user ID.
536         This privilege is interpreted only if the system is configured
537         with Trusted Extensions.

539 privilege PRIV_WIN_DEVICES

541         Allows a process to perform operations on window input devices.
542         Allows a process to get and set keyboard and pointer controls.
543         Allows a process to modify pointer button and key mappings.
544         This privilege is interpreted only if the system is configured
545         with Trusted Extensions.

547 privilege PRIV_WIN_DGA

549         Allows a process to use the direct graphics access (DGA) X protocol
550         extensions. Direct process access to the frame buffer is still
551         required. Thus the process must have MAC and DAC privileges that
552         allow access to the frame buffer, or the frame buffer must be
553         allocated to the process.
554         This privilege is interpreted only if the system is configured
555         with Trusted Extensions.

557 privilege PRIV_WIN_DOWNGRADE_SL

559         Allows a process to set the sensitivity label of a window resource
560         to a sensitivity label that does not dominate the existing
561         sensitivity label.
562         This privilege is interpreted only if the system is configured
563         with Trusted Extensions.

565 privilege PRIV_WIN_FONTPATH

567         Allows a process to set a font path.
568         This privilege is interpreted only if the system is configured
569         with Trusted Extensions.

571 privilege PRIV_WIN_MAC_READ

573         Allows a process to read from a window resource whose sensitivity
574         label is not equal to the process sensitivity label.
575         This privilege is interpreted only if the system is configured
576         with Trusted Extensions.

578 privilege PRIV_WIN_MAC_WRITE

580         Allows a process to create a window resource whose sensitivity
581         label is not equal to the process sensitivity label.
582         A newly created window property is created with the window's
583         sensitivity label.
584         This privilege is interpreted only if the system is configured
585         with Trusted Extensions.

587 privilege PRIV_WIN_SELECTION
```



```
589         Allows a process to request inter-window data moves without the
590         intervention of the selection confirmer.
591         This privilege is interpreted only if the system is configured
592         with Trusted Extensions.

594 privilege PRIV_WIN_UPGRADE_SL

596         Allows a process to set the sensitivity label of a window
597         resource to a sensitivity label that dominates the existing
598         sensitivity label.
599         This privilege is interpreted only if the system is configured
600         with Trusted Extensions.

602 privilege PRIV_XVM_CONTROL

604         Allows a process access to the xVM(5) control devices for
605         managing guest domains and the hypervisor. This privilege is
606         used only if booted into xVM on x86 platforms.

608 set PRIV_EFFECTIVE

610         Set of privileges currently in effect.

612 set PRIV_INHERITABLE

614         Set of privileges that comes into effect on exec.

616 set PRIV_PERMITTED

618         Set of privileges that can be put into the effective set without
619         restriction.

621 set PRIV_LIMIT

623         Set of privileges that determines the absolute upper bound of
624         privileges this process and its off-spring can obtain.
```

```
**********************************************************
   11570 Mon Dec 28 20:02:42 2015
new/usr/src/uts/common/os/privs.awk
uts: add a concept of a 'default' set of privileges, separate from 'basic'
uts: give privilege macros more sensible names
**********************************************************
______unchanged_portion_omitted_

 100 #
 101 # Privilege strings are represented as lower case strings;
 102 # PRIV_ is stripped from the strings.
 103 #
 104 /^([A-Za-z]* )?privilege / {
 105         if (NF == 3) {
 106                 key = toupper($1)
 107                 priv = toupper($3)
 108                 if (set[key] != "")
 109                         set[key] = set[key] ";"
 110                 set[key] = set[key] "\\\n\t\tPRIV_ADDSET((set), " priv ")"
 110                 set[key] = set[key] "\\\n\t\tPRIV_ASSERT((set), " priv ")"
 111         } else {
 112                 priv = toupper($2);
 113         }
 114         privs[npriv] = tolower(substr(priv, 6));
 115         inset = 0
 116         inpriv = 1

 118         privind[npriv] = privbytes;

 120         tabs = (32 - length(priv) - 1)/8
 121         # length + \0 - PRIV_
 122         privbytes += length(priv) - 4
 123         pdef[npriv] = "#define\t" priv substr("\t\t\t\t\t", 1, tabs)

 125         npriv++
 126         next
 127 }
______unchanged_portion_omitted_

 213 END     {

 215         if (!pubhfile && !privhfile && !cfile && !pnamesfile) {
 216                 print "Output file parameter not set" > "/dev/stderr"
 217                 exit 1
 218         }

 220         setsize = int((npriv + slack)/(8 * 4)) + 1
 221         maxnpriv = setsize * 8 * 4
 222         # Assume allocated privileges are on average "NSDQ" bytes larger.
 223         maxprivbytes = int((privbytes / npriv + 5.5)) * (maxnpriv - npriv)
 224         maxprivbytes += privbytes

 226         if (cfile) {
 227                 print "\n" > cfile
 228                 print pragma "\n"> cfile
 229                 print "#include <sys/types.h>" > cfile
 230                 print "#include <sys/priv_const.h>" > cfile
 231                 print "#include <sys/priv_impl.h>" > cfile
 232                 print "#include <sys/priv.h>" > cfile
 233                 print "#include <sys/sysmacros.h>" > cfile
 234                 print "\n" > cfile
 235                 #
 236                 # Create the entire priv info structure here.
 237                 # When adding privileges, the kernel needs to update
 238                 # too many fields as the number of privileges is kept in
 239                 # many places.
```



```
 240                 #
 241                 print \
 242                     "static struct _info {\n" \
 243                     "   priv_impl_info_t        impl_info;\n" \
 244                     "   priv_info_t             settype;\n" \
 245                     "   int                     nsets;\n" \
 246                     "   const char              sets[" setbytes "];\n" \
 247                     "   priv_info_t             privtype;\n" \
 248                     "   int                     nprivs;\n" \
 249                     "   char                    privs[" maxprivbytes "];\n" \
 250                     "   priv_info_t             sysset;\n" \
 251                     "   priv_set_t              basicset;\n" \
 252                     "   priv_info_t             defset;\n" \
 253                     "   priv_set_t              defaultset;\n" \
 254 #endif /* ! codereview */
 255                     "} info = {\n" \
 256                     "   { sizeof (priv_impl_info_t), 0, PRIV_NSET, " \
 257                         "PRIV_SETSIZE, " npriv ",\n" \
 258                         "\t\tsizeof (priv_info_uint_t),\n" \
 259                         "\t\tsizeof (info) - sizeof (info.impl_info)},\n" \
 260                     "   { PRIV_INFO_SETNAMES,\n" \
 261                     "       offsetof(struct _info, privtype) - " \
 262                     "offsetof(struct _info, settype)},\n\tPRIV_NSET," > cfile

 264                 sep = "\t\""
 265                 len = 9;
 266                 for (i = 0; i < nset; i++) {
 267                         if (len + length(sets[i]) > 80) {
 268                                 sep = "\\0\"\n\t\""
 269                                 len = 9
 270                         }
 271                         printf sep sets[i]  > cfile
 272                         len += length(sets[i]) + length(sep);
 273                         sep = "\\0"
 274                 }
 275                 print "\\0\"," > cfile

 277                 print "\t{ PRIV_INFO_PRIVNAMES,\n\t    " \
 278                         "offsetof(struct _info, sysset) - " \
 279                         "offsetof(struct _info, privtype)},\n\t" npriv "," \
 280                         > cfile

 282                 sep = "\t\""
 283                 len = 9;
 284                 for (i = 0; i < npriv; i++) {
 285                         if (len + length(privs[i]) > 80) {
 286                                 sep = "\\0\"\n\t\""
 287                                 len = 9
 288                         }
 289                         printf sep privs[i]  > cfile
 290                         len += length(privs[i]) + length(sep);
 291                         sep = "\\0"
 292                 }
 293                 print "\\0\"," > cfile

 295                 print "\t{ PRIV_INFO_BASICPRIVS, offsetof (struct _info, defset)
 252                 print "\t{ PRIV_INFO_BASICPRIVS, sizeof (info) - " \
 296                     "offsetof(struct _info, sysset)},"  > cfile
 297                 print "\t{ 0 },\n" > cfile
 298                 print "\t{ PRIV_INFO_DEFAULTPRIVS, sizeof (info) - " \
 299                     "offsetof(struct _info, defset)}" > cfile
 300 #endif /* ! codereview */

 302                 print "};\n" > cfile

 304                 print "\nconst char *priv_names[" maxnpriv "] =\n{" > cfile
```

```
305                 for (i = 0; i < npriv; i++)
306                         print "\t&info.privs[" privind[i] "]," > cfile

308                 print "};\n" > cfile

310                 print "\nconst char *priv_setnames[" nset "] =\n{" > cfile
311                 for (i = 0; i < nset; i++)
312                         print "\t&info.sets[" setind[i] "]," > cfile

314                 print "};\n" > cfile

316                 print "int nprivs = " npriv ";" > cfile
317                 print "int privbytes = " privbytes ";" > cfile
318                 print "int maxprivbytes = " maxprivbytes ";" > cfile
319                 print "size_t privinfosize = sizeof (info);" > cfile
320                 print "char *priv_str = info.privs;" > cfile
321                 print "priv_set_t *priv_basic = &info.basicset;" > cfile
322                 print "priv_set_t *priv_default = &info.defaultset;" > cfile
323 #endif /* ! codereview */
324                 print "priv_impl_info_t *priv_info = &info.impl_info;" > cfile
325                 print "priv_info_names_t *priv_ninfo = " \
326                         "(priv_info_names_t *)&info.privtype;" > cfile
327                 close(cfile)
328         }

330         # Kernel private
331         if (privhfile) {
332                 print "#ifndef _SYS_PRIV_CONST_H" > privhfile
333                 print "#define\t_SYS_PRIV_CONST_H\n" > privhfile
334                 print pragma "\n"> privhfile
335                 print "\n#include <sys/types.h>\n\n" > privhfile
336                 print "#ifdef __cplusplus\nextern \"C\" {\n#endif\n" > privhfile

338                 print "#if defined(_KERNEL) || defined(_KMEMUSER)" > privhfile
339                 print "#define\tPRIV_NSET\t\t\t  " nset > privhfile
340                 print "#define\tPRIV_SETSIZE\t\t\t  " setsize > privhfile
341                 print "#endif\n\n#ifdef _KERNEL" > privhfile
342                 print "#define\t__PRIV_CONST_IMPL\n" > privhfile
343                 print "extern const char *priv_names[];" > privhfile
344                 print "extern const char *priv_setnames[];" > privhfile

346                 print "extern int nprivs;" > privhfile
347                 print "extern int privbytes;" > privhfile
348                 print "extern int maxprivbytes;" > privhfile
349                 print "extern size_t privinfosize;" > privhfile
350                 print "extern char *priv_str;" > privhfile
351                 print "extern struct priv_set *priv_basic;" > privhfile
352                 print "extern struct priv_set *priv_default;" > privhfile
353 #endif /* ! codereview */
354                 print "extern struct priv_impl_info *priv_info;" > privhfile
355                 print "extern struct priv_info_names *priv_ninfo;" > privhfile

357                 print "\n/* Privileges */" > privhfile
358
359                 for (i = 0; i < npriv; i++)
360                         print pdef[i] sprintf("%3d", i) > privhfile

362                 print "\n/* Privilege sets */" > privhfile
363                 for (i = 0; i < nset; i++)
364                         print sdef[i] sprintf("%3d", i) > privhfile

366                 print "\n#define\tMAX_PRIVILEGE\t\t\t "  setsize * 32 \
367                         > privhfile

369                 # Special privilege categories.
370                 for (s in set)
```



```
371                         print "\n#define\tPRIV_" s "_ADDSET(set)" set[s] \
254                         print "\n#define\tPRIV_" s "_ASSERT(set)" set[s] \
372                                 > privhfile

374                 print "\n#endif /* _KERNEL */" > privhfile
375                 print "\n#ifdef __cplusplus\n}\n#endif" > privhfile
376                 print "\n#endif /* _SYS_PRIV_CONST_H */" > privhfile
377                 close(privhfile)
378         }

380         if (pubhfile) {
381                 cast="(const char *)"
382                 print "#ifndef _SYS_PRIV_NAMES_H" > pubhfile
383                 print "#define\t_SYS_PRIV_NAMES_H\n" > pubhfile

385                 print pragma "\n" > pubhfile
386                 print "#ifdef __cplusplus\nextern \"C\" {\n#endif\n" > pubhfile

388                 print "#ifndef __PRIV_CONST_IMPL" > pubhfile
389                 print "/*\n * Privilege names\n */" > pubhfile
390                 for (i = 0; i < npriv; i++) {
391                         print "/*\n" privcmt[i] " */" > pubhfile
392                         print pdef[i] cast "\"" privs[i] "\")\n" > pubhfile
393                 }

395                 print "" > pubhfile

397                 print "/*\n * Privilege set names\n */" > pubhfile
398                 for (i = 0; i < nset; i++) {
399                         print "/*\n" setcmt[i] " */" > pubhfile
400                         print sdef[i] cast "\"" sets[i] "\")\n" > pubhfile
401                 }

403                 print "\n#endif /* __PRIV_CONST_IMPL */" > pubhfile
404                 print "\n#ifdef __cplusplus\n}\n#endif" > pubhfile
405                 print "\n#endif /* _SYS_PRIV_NAMES_H */" > pubhfile
406                 close(pubhfile)
407         }

409         if (pnamesfile) {
410                 print pnamescmt > pnamesfile
411                 for (i = 0; i < npriv; i++) {
412                         print privs[i] > pnamesfile
413                         print privncmt[i] > pnamesfile
414                 }
415         }

417 }
```

*_____unchanged_portion_omitted_*

```
**********************************************************
    7325 Mon Dec 28 20:02:43 2015
new/usr/src/uts/common/sys/priv.h
uts: add a concept of a 'default' set of privileges, separate from 'basic'
**********************************************************
______unchanged_portion_omitted_

 179 /*
 180  * Privilege information types.
 181  */
 182 #define PRIV_INFO_SETNAMES              0x0001
 183 #define PRIV_INFO_PRIVNAMES             0x0002
 184 #define PRIV_INFO_BASICPRIVS            0x0003
 185 #define PRIV_INFO_FLAGS                 0x0004
 186 #define PRIV_INFO_DEFAULTPRIVS          0x0005
 187 #endif /* ! codereview */

 189 /*
 190  * Special "privileges" used to indicate special conditions in privilege
 191  * debugging/tracing code.
 192  */
 193 #define PRIV_ALL                        (-1)    /* All privileges required */
 194 #define PRIV_MULTIPLE                   (-2)    /* More than one */
 195 #define PRIV_NONE                       (-3)    /* No value */
 196 #define PRIV_ALLZONE                    (-4)    /* All privileges in zone */
 197 #define PRIV_GLOBAL                     (-5)    /* Must be in global zone */

 199 #ifdef _KERNEL

 201 #define PRIV_ALLOC                      0x1

 203 extern int priv_debug;
 204 extern int priv_basic_test;

 206 struct proc;
 207 struct prpriv;
 208 struct cred;

 210 extern int priv_prgetprivsize(struct prpriv *);
 211 extern void cred2prpriv(const struct cred *, struct prpriv *);
 212 extern int priv_pr_spriv(struct proc *, struct prpriv *, const struct cred *);

 214 extern priv_impl_info_t *priv_hold_implinfo(void);
 215 extern void priv_release_implinfo(void);
 216 extern size_t priv_get_implinfo_size(void);
 217 extern const priv_set_t *priv_getset(const struct cred *, int);
 218 extern void priv_getinfo(const struct cred *, void *);
 219 extern int priv_getbyname(const char *, uint_t);
 220 extern int priv_getsetbyname(const char *, int);
 221 extern const char *priv_getbynum(int);
 222 extern const char *priv_getsetbynum(int);

 224 extern void priv_emptyset(priv_set_t *);
 225 extern void priv_fillset(priv_set_t *);
 226 extern void priv_addset(priv_set_t *, int);
 227 extern void priv_delset(priv_set_t *, int);
 228 extern boolean_t priv_ismember(const priv_set_t *, int);
 229 extern boolean_t priv_isemptyset(const priv_set_t *);
 230 extern boolean_t priv_isfullset(const priv_set_t *);
 231 extern boolean_t priv_isequalset(const priv_set_t *, const priv_set_t *);
 232 extern boolean_t priv_issubset(const priv_set_t *, const priv_set_t *);
 233 extern int priv_proc_cred_perm(const struct cred *, struct proc *,
 234         struct cred **, int);
 235 extern void priv_intersect(const priv_set_t *, priv_set_t *);
 236 extern void priv_union(const priv_set_t *, priv_set_t *);
 237 extern void priv_inverse(priv_set_t *);
```



```
 239 extern void priv_set_PA(cred_t *);
 240 extern void priv_adjust_PA(cred_t *);
 241 extern void priv_reset_PA(cred_t *, boolean_t);
 242 extern boolean_t priv_can_clear_PA(const cred_t *);

 244 extern int setpflags(uint_t, uint_t, cred_t *);
 245 extern uint_t getpflags(uint_t, const cred_t *);

 247 #endif /* _KERNEL */

 249 #ifdef  __cplusplus
 250 }
 251 #endif

 253 #endif  /* _SYS_PRIV_H */
```

```
**********************************************************
    3628 Mon Dec 28 20:02:44 2015
new/usr/src/uts/common/sys/priv_impl.h
uts: give privilege macros more sensible names
**********************************************************
```

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License, Version 1.0 only
 * (the "License").  You may not use this file except in compliance
 * with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright 2003 Sun Microsystems, Inc.  All rights reserved.
 * Use is subject to license terms.
 */

#ifndef _SYS_PRIV_IMPL_H
#define _SYS_PRIV_IMPL_H

#pragma ident   "%Z%%M% %I%     %E% SMI"

#include <sys/priv_const.h>
#include <sys/priv.h>

#ifdef  __cplusplus
extern "C" {
#endif

#if defined(_KERNEL) || defined(_KMEMUSER)
/*
 * priv_set_t is a structure holding a set of privileges
 */

struct priv_set {
        priv_chunk_t pbits[PRIV_SETSIZE];
};
```

*unchanged_portion_omitted_*

```
#endif

#ifdef _KERNEL

extern priv_set_t *priv_basic;
extern priv_set_t priv_unsafe;
extern priv_set_t priv_fullset;
extern void priv_init(void);

/* The CR_PRIVS macro is defined in <sys/cred_impl.h> */
#define CR_EPRIV(c)     (CR_PRIVS(c)->crprivs[PRIV_EFFECTIVE])
#define CR_IPRIV(c)     (CR_PRIVS(c)->crprivs[PRIV_INHERITABLE])
#define CR_PPRIV(c)     (CR_PRIVS(c)->crprivs[PRIV_PERMITTED])
```



```
#define CR_LPRIV(c)     (CR_PRIVS(c)->crprivs[PRIV_LIMIT])

#define CR_FLAGS(c)     (CR_PRIVS(c)->crpriv_flags)

#define PRIV_SETBYTES   (PRIV_NSET * PRIV_SETSIZE * sizeof (priv_chunk_t))

#define PRIV_EISAWARE(c) ((CR_FLAGS(c) & PRIV_AWARE) || (c)->cr_uid != 0)
#define PRIV_PISAWARE(c) ((CR_FLAGS(c) & PRIV_AWARE) || \
                                ((c)->cr_uid != 0 && (c)->cr_suid != 0 && \
                                (c)->cr_ruid != 0))

#define CR_OEPRIV(c)    (*(PRIV_EISAWARE(c) ? &CR_EPRIV(c) : &CR_LPRIV(c)))
#define CR_OPPRIV(c)    (*(PRIV_PISAWARE(c) ? &CR_PPRIV(c) : &CR_LPRIV(c)))

#define PRIV_VALIDSET(s)        ((s) >= 0 && (s) < PRIV_NSET)
#define PRIV_VALIDOP(op)        ((op) >= PRIV_ON && (op) <= PRIV_SET)

#define PRIV_FULLSET            &priv_fullset   /* Require full set */

/*
 * Privilege macros bits manipulation macros; DEBUG kernels will
 * ASSERT() that privileges are not out of range.
 */
#ifndef NBBY
#define NBBY            8
#endif

#define __NBWRD         (NBBY * sizeof (priv_chunk_t))

#define privmask(n)     (1U << ((__NBWRD - 1) - ((n) % __NBWRD)))
#define privword(n)     ((n)/__NBWRD)

/*
 * PRIV_ADDSET(a, b) sets privilege "b" in privilege set "a".
 * PRIV_DELSET(a, b) clears privilege "b" in privilege set "a".
 * PRIV_ISMEMBER(a, b) tests if privilege 'b' is asserted in privilege set 'a'.
 * PRIV_ASSERT(a, b) sets privilege "b" in privilege set "a".
 * PRIV_CLEAR(a,b) clears privilege "b" in privilege set "a".
 * PRIV_ISASSERT tests if privilege 'b' is asserted in privilege set 'a'.
 */

#define __PRIV_ADDSET(a, b)     ((a)->pbits[privword(b)] |= privmask(b))
#define __PRIV_DELSET(a, b)     ((a)->pbits[privword(b)] &= ~privmask(b))
#define __PRIV_ISMEMBER(a, b)   ((a)->pbits[privword(b)] & privmask(b))
#define __PRIV_ASSERT(a, b)     ((a)->pbits[privword(b)] |= privmask(b))
#define __PRIV_CLEAR(a, b)      ((a)->pbits[privword(b)] &= ~privmask(b))
#define __PRIV_ISASSERT(a, b)   ((a)->pbits[privword(b)] & privmask(b))

#ifdef DEBUG
#define PRIV_DELSET(a, b)       priv_delset((a), (b))
#define PRIV_ADDSET(a, b)       priv_addset((a), (b))
#define PRIV_ISMEMBER(a, b)     priv_ismember((a), (b))
#define PRIV_CLEAR(a, b)        priv_delset((a), (b))
#define PRIV_ASSERT(a, b)       priv_addset((a), (b))
#define PRIV_ISASSERT(a, b)     priv_ismember((a), (b))
#else
#define PRIV_DELSET(a, b)       __PRIV_DELSET((a), (b))
#define PRIV_ADDSET(a, b)       __PRIV_ADDSET((a), (b))
#define PRIV_ISMEMBER(a, b)     __PRIV_ISMEMBER((a), (b))
#define PRIV_CLEAR(a, b)        __PRIV_CLEAR((a), (b))
#define PRIV_ASSERT(a, b)       __PRIV_ASSERT((a), (b))
#define PRIV_ISASSERT(a, b)     __PRIV_ISASSERT((a), (b))
#endif

#endif /* _KERNEL */
```

```
 118 #ifdef  __cplusplus
 119 }
```

_____***unchanged_portion_omitted_***

```
**********************************************************
    3459 Mon Dec 28 20:02:46 2015
new/usr/src/uts/common/sys/tsol/priv.h
uts: give privilege macros more sensible names
**********************************************************
```

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License (the "License").
 * You may not use this file except in compliance with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright 2006 Sun Microsystems, Inc.  All rights reserved.
 * Use is subject to license terms.
 */

#ifndef _SYS_TSOL_PRIV_H
#define _SYS_TSOL_PRIV_H

#pragma ident   "%Z%%M% %I%     %E% SMI"

#include <sys/priv.h>

#ifdef  __cplusplus
extern "C" {
#endif

typedef enum priv_ftype {
        PRIV_ALLOWED,
        PRIV_FORCED
} priv_ftype_t;

/*
 * Privilege macros.
 *
 * These names are here for compatibility reasons, and thus do not match
 * priv_impl.h
#endif /* ! codereview */
 */

/*
 * PRIV_ASSERT(a, b) setst.privilege "b" in privilege set "a".
 */
#define PRIV_ASSERT(a, b) (priv_addset(a, b))

/*
 * PRIV_CLEAR(a,b) clearst.privilege "b" in privilege set "a".
 */
#define PRIV_CLEAR(a, b) (priv_delset(a, b))

/*
 * PRIV_EQUAL(set_a, set_b) is true if set_a and set_b are identical.
```



```
 */
#define PRIV_EQUAL(a, b) (priv_isequalset(a, b))
#define PRIV_EMPTY(a) (priv_emptyset(a))
#define PRIV_FILL(a) (priv_fillset(a))

/*
 * PRIV_ISASSERT tests if privilege 'b' is asserted in privilege set 'a'.
 */
#define PRIV_ISASSERT(a, b) (priv_ismember(a, b))
#define PRIV_ISEMPTY(a) (priv_isemptyset(a))
#define PRIV_ISFULL(a) (priv_isfullset(a))

/*
 * This macro returns 1 if all privileges asserted in privilege set "a"
 * are also asserted in privilege set "b" (i.e. if a is a subset of b)
 */
#define PRIV_ISSUBSET(a, b) (priv_issubset(a, b))

/*
 * Takes intersection of "a" and "b" and stores in "b".
 */
#define PRIV_INTERSECT(a, b) (priv_intersect(a, b))

/*
 * Replaces "a" with inverse of "a".
 */
#define PRIV_INVERSE(a)  (priv_inverse(a))

/*
 * Takes union of "a" and "b" and stores in "b".
 */
#define PRIV_UNION(a, b) (priv_union(a, b))


#define PRIV_FILE_UPGRADE_SL    ((const char *)"file_upgrade_sl")
#define PRIV_FILE_DOWNGRADE_SL  ((const char *)"file_downgrade_sl")
#
#define PRIV_PROC_AUDIT_TCB     ((const char *)"proc_audit")
#define PRIV_PROC_AUDIT_APPL    ((const char *)"proc_audit")
#
#define PRIV_SYS_TRANS_LABEL    ((const char *)"sys_trans_label")
#define PRIV_WIN_COLORMAP       ((const char *)"win_colormap")
#define PRIV_WIN_CONFIG         ((const char *)"win_config")
#define PRIV_WIN_DAC_READ       ((const char *)"win_dac_read")
#define PRIV_WIN_DAC_WRITE      ((const char *)"win_dac_write")
#define PRIV_WIN_DGA            ((const char *)"win_dga")
#define PRIV_WIN_DEVICES        ((const char *)"win_devices")
#define PRIV_WIN_DOWNGRADE_SL   ((const char *)"win_downgrade_sl")
#define PRIV_WIN_FONTPATH       ((const char *)"win_fontpath")
#define PRIV_WIN_MAC_READ       ((const char *)"win_mac_read")
#define PRIV_WIN_MAC_WRITE      ((const char *)"win_mac_write")
#define PRIV_WIN_SELECTION      ((const char *)"win_selection")
#define PRIV_WIN_UPGRADE_SL     ((const char *)"win_upgrade_sl")

#ifdef  __cplusplus
}
#endif

#endif  /* _SYS_TSOL_PRIV_H */
```

```
**********************************************************
   10217 Mon Dec 28 20:02:47 2015
new/usr/src/uts/common/syscall/ppriv.c
uts: give privilege macros more sensible names
**********************************************************
```

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License (the "License").
 * You may not use this file except in compliance with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright (c) 2003, 2010, Oracle and/or its affiliates. All rights reserved.
 */

#include <sys/param.h>
#include <sys/types.h>
#include <sys/sysmacros.h>
#include <sys/systm.h>
#include <sys/cred_impl.h>
#include <sys/errno.h>
#include <sys/klpd.h>
#include <sys/proc.h>
#include <sys/priv_impl.h>
#include <sys/policy.h>
#include <sys/ddi.h>
#include <sys/thread.h>
#include <sys/cmn_err.h>
#include <c2/audit.h>

/*
 * System call support for manipulating privileges.
 *
 *
 * setppriv(2) - set process privilege set
 * getppriv(2) - get process privilege set
 * getprivimplinfo(2) - get process privilege implementation information
 * setpflags(2) - set process (privilege) flags
 * getpflags(2) - get process (privilege) flags
 */

/*
 * setppriv (priv_op_t, priv_ptype_t, priv_set_t)
 */
static int
setppriv(priv_op_t op, priv_ptype_t type, priv_set_t *in_pset)
{
        priv_set_t      pset, *target;
        cred_t          *cr, *pcr;
        proc_t          *p;
        boolean_t       donocd = B_FALSE;
```



```
        if (!PRIV_VALIDSET(type) || !PRIV_VALIDOP(op))
                return (set_errno(EINVAL));

        if (copyin(in_pset, &pset, sizeof (priv_set_t)))
                return (set_errno(EFAULT));

        p = ttoproc(curthread);
        cr = cralloc();
        mutex_enter(&p->p_crlock);

retry:
        pcr = p->p_cred;

        if (AU_AUDITING())
                audit_setppriv(op, type, &pset, pcr);

        /*
         * Filter out unallowed request (bad op and bad type)
         */
        switch (op) {
        case PRIV_ON:
        case PRIV_SET:
                /*
                 * Turning on privileges; the limit set cannot grow,
                 * other sets can but only as long as they remain subsets
                 * of P.  Only immediately after exec holds that P <= L.
                 */
                if (type == PRIV_LIMIT &&
                    !priv_issubset(&pset, &CR_LPRIV(pcr))) {
                        mutex_exit(&p->p_crlock);
                        crfree(cr);
                        return (set_errno(EPERM));
                }
                if (!priv_issubset(&pset, &CR_OPPRIV(pcr)) &&
                    !priv_issubset(&pset, priv_getset(pcr, type))) {
                        mutex_exit(&p->p_crlock);
                        /* Policy override should not grow beyond L either */
                        if (type != PRIV_INHERITABLE ||
                            !priv_issubset(&pset, &CR_LPRIV(pcr)) ||
                            secpolicy_require_privs(CRED(), &pset) != 0) {
                                crfree(cr);
                                return (set_errno(EPERM));
                        }
                        mutex_enter(&p->p_crlock);
                        if (pcr != p->p_cred)
                                goto retry;
                        donocd = B_TRUE;
                }
                break;

        case PRIV_OFF:
                /* PRIV_OFF is always allowed */
                break;
        }

        /*
         * OK! everything is cool.
         * Do cred COW.
         */
        crcopy_to(pcr, cr);

        /*
         * If we change the effective, permitted or limit set, we attain
         * "privilege awareness".
         */
        if (type != PRIV_INHERITABLE)
```

```
128                 priv_set_PA(cr);

130         target = &(CR_PRIVS(cr)->crprivs[type]);

132         switch (op) {
133         case PRIV_ON:
134                 priv_union(&pset, target);
135                 break;
136         case PRIV_OFF:
137                 priv_inverse(&pset);
138                 priv_intersect(target, &pset);

140                 /*
141                  * Fall-thru to set target and change other process
142                  * privilege sets.
143                  */
144                 /*FALLTHRU*/

146         case PRIV_SET:
147                 *target = pset;

149                 /*
150                  * Take privileges no longer permitted out
151                  * of other effective sets as well.
152                  * Limit set is enforced at exec() time.
153                  */
154                 if (type == PRIV_PERMITTED)
155                         priv_intersect(&pset, &CR_EPRIV(cr));
156                 break;
157         }

159         /*
160          * When we give up privileges not in the inheritable set,
161          * set SNOCD if not already set; first we compute the
162          * privileges removed from P using Diff = (~P') & P
163          * and then we check whether the removed privileges are
164          * a subset of I.  If we retain uid 0, all privileges
165          * are required anyway so don't set SNOCD.
166          */
167         if (type == PRIV_PERMITTED && (p->p_flag & SNOCD) == 0 &&
168             cr->cr_uid != 0 && cr->cr_ruid != 0 && cr->cr_suid != 0) {
169                 priv_set_t diff = CR_OPPRIV(cr);
170                 priv_inverse(&diff);
171                 priv_intersect(&CR_OPPRIV(pcr), &diff);
172                 donocd = !priv_issubset(&diff, &CR_IPRIV(cr));
173         }

175         p->p_cred = cr;
176         mutex_exit(&p->p_crlock);

178         if (donocd) {
179                 mutex_enter(&p->p_lock);
180                 p->p_flag |= SNOCD;
181                 mutex_exit(&p->p_lock);
182         }

184         /*
185          * The basic_test privilege should not be removed from E;
186          * if that has happened, then some programmer typically set the E/P to
187          * empty. That is not portable.
188          */
189         if ((type == PRIV_EFFECTIVE || type == PRIV_PERMITTED) &&
190             priv_basic_test >= 0 && !PRIV_ISMEMBER(target, priv_basic_test)) {
190             priv_basic_test >= 0 && !PRIV_ISASSERT(target, priv_basic_test)) {
191                 proc_t *p = curproc;
192                 pid_t pid = p->p_pid;
```



```
193                 char *fn = PTOU(p)->u_comm;

195                 cmn_err(CE_WARN, "%s[%d]: setppriv: basic_test privilege "
196                     "removed from E/P", fn, pid);
197         }

199         crset(p, cr);           /* broadcast to process threads */

201         return (0);
202 }
```

_____***unchanged_portion_omitted_***