

```

*****
205489 Thu Jan 17 14:44:29 2019
new/usr/src/lib/libzonecfg/common/libzonecfg.c
10109 libzonecfg needs a smatch fix
*****
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 */

22 /*
23  * Copyright 2014 Gary Mills
24  * Copyright (c) 2003, 2010, Oracle and/or its affiliates. All rights reserved.
25  * Copyright (c) 2018, Joyent, Inc.
26  * Copyright 2015 Nexenta Systems, Inc. All rights reserved.
27 */

29 #include <libsysevent.h>
30 #include <pthread.h>
31 #include <stdlib.h>
32 #include <errno.h>
33 #include <fnmatch.h>
34 #include <strings.h>
35 #include <unistd.h>
36 #include <assert.h>
37 #include <libgen.h>
38 #include <libintl.h>
39 #include <alloca.h>
40 #include <ctype.h>
41 #include <sys/acl.h>
42 #include <sys/stat.h>
43 #include <sys/brand.h>
44 #include <sys/mntio.h>
45 #include <sys/mnttab.h>
46 #include <sys/nvpair.h>
47 #include <sys/types.h>
48 #include <sys/sockio.h>
49 #include <sys/systeminfo.h>
50 #include <ftw.h>
51 #include <pool.h>
52 #include <libscf.h>
53 #include <libproc.h>
54 #include <sys/priocntl.h>
55 #include <libutil.h>
56 #include <wait.h>
57 #include <bsm/adt.h>
58 #include <auth_attr.h>
59 #include <auth_list.h>
60 #include <secdb.h>
61 #include <user_attr.h>

```

```

62 #include <prof_attr.h>
63
64 #include <arpa/inet.h>
65 #include <netdb.h>
66
67 #include <libxml/xmlmemory.h>
68 #include <libxml/parser.h>
69
70 #include <libdevinfo.h>
71 #include <uuid/uuid.h>
72 #include <dirent.h>
73 #include <libbrand.h>
74
75 #include <libzonecfg.h>
76 #include "zonecfg_impl.h"
77
78 #define _PATH_TMPFILE "/zonecfg.XXXXXX"
79 #define ZONE_CB_RETRY_COUNT 10
80 #define ZONE_EVENT_PING_SUBCLASS "ping"
81 #define ZONE_EVENT_PING_PUBLISHER "solaris"
82
83 /* Hard-code the DTD element/attribute/entity names just once, here. */
84 #define DTD_ELEM_ATTR (const xmlChar *) "attr"
85 #define DTD_ELEM_COMMENT (const xmlChar *) "comment"
86 #define DTD_ELEM_DEVICE (const xmlChar *) "device"
87 #define DTD_ELEM_FS (const xmlChar *) "filesystem"
88 #define DTD_ELEM_FSOPTION (const xmlChar *) "fsoption"
89 #define DTD_ELEM_NET (const xmlChar *) "network"
90 #define DTD_ELEM_RCTL (const xmlChar *) "rctl"
91 #define DTD_ELEM_RCTLVALUE (const xmlChar *) "rctl-value"
92 #define DTD_ELEM_ZONE (const xmlChar *) "zone"
93 #define DTD_ELEM_DATASET (const xmlChar *) "dataset"
94 #define DTD_ELEM_TMPPPOOL (const xmlChar *) "tmp_pool"
95 #define DTD_ELEM_PSET (const xmlChar *) "pset"
96 #define DTD_ELEM_MCAP (const xmlChar *) "mcap"
97 #define DTD_ELEM_PACKAGE (const xmlChar *) "package"
98 #define DTD_ELEM_OBSOLETES (const xmlChar *) "obsoletes"
99 #define DTD_ELEM_DEV_PERM (const xmlChar *) "dev-perm"
100 #define DTD_ELEM_ADMIN (const xmlChar *) "admin"
101 #define DTD_ELEM_SECFLAGS (const xmlChar *) "security-flags"
102
103 #define DTD_ATTR_ACTION (const xmlChar *) "action"
104 #define DTD_ATTR_ADDRESS (const xmlChar *) "address"
105 #define DTD_ATTR_ALLOWED_ADDRESS (const xmlChar *) "allowed-address"
106 #define DTD_ATTR_AUTOBOOT (const xmlChar *) "autoboot"
107 #define DTD_ATTR_IPTYPE (const xmlChar *) "ip-type"
108 #define DTD_ATTR_DEFROUTER (const xmlChar *) "defrouter"
109 #define DTD_ATTR_DIR (const xmlChar *) "directory"
110 #define DTD_ATTR_LIMIT (const xmlChar *) "limit"
111 #define DTD_ATTR_LIMITPRIV (const xmlChar *) "limitpriv"
112 #define DTD_ATTR_BOOTTARGS (const xmlChar *) "bootargs"
113 #define DTD_ATTR_SCHED (const xmlChar *) "scheduling-class"
114 #define DTD_ATTR_MATCH (const xmlChar *) "match"
115 #define DTD_ATTR_NAME (const xmlChar *) "name"
116 #define DTD_ATTR_PHYSICAL (const xmlChar *) "physical"
117 #define DTD_ATTR_POOL (const xmlChar *) "pool"
118 #define DTD_ATTR_PRIV (const xmlChar *) "priv"
119 #define DTD_ATTR_RAW (const xmlChar *) "raw"
120 #define DTD_ATTR_SPECIAL (const xmlChar *) "special"
121 #define DTD_ATTR_TYPE (const xmlChar *) "type"
122 #define DTD_ATTR_VALUE (const xmlChar *) "value"
123 #define DTD_ATTR_ZONEPATH (const xmlChar *) "zonepath"
124 #define DTD_ATTR_NCPU_MIN (const xmlChar *) "ncpu_min"
125 #define DTD_ATTR_NCPU_MAX (const xmlChar *) "ncpu_max"
126 #define DTD_ATTR_IMPORTANCE (const xmlChar *) "importance"
127 #define DTD_ATTR_PHYSCAP (const xmlChar *) "physcap"

```

```

128 #define DTD_ATTR_VERSION      (const xmlChar *) "version"
129 #define DTD_ATTR_ID          (const xmlChar *) "id"
130 #define DTD_ATTR_UID         (const xmlChar *) "uid"
131 #define DTD_ATTR_GID         (const xmlChar *) "gid"
132 #define DTD_ATTR_MODE        (const xmlChar *) "mode"
133 #define DTD_ATTR_ACL         (const xmlChar *) "acl"
134 #define DTD_ATTR_BRAND       (const xmlChar *) "brand"
135 #define DTD_ATTR_HOSTID      (const xmlChar *) "hostid"
136 #define DTD_ATTR_USER        (const xmlChar *) "user"
137 #define DTD_ATTR_AUTHS       (const xmlChar *) "auths"
138 #define DTD_ATTR_FS_ALLOWED  (const xmlChar *) "fs-allowed"
139 #define DTD_ATTR_DEFAULT     (const xmlChar *) "default"
140 #define DTD_ATTR_LOWER       (const xmlChar *) "lower"
141 #define DTD_ATTR_UPPER       (const xmlChar *) "upper"

144 #define DTD_ENTITY_BOOLEAN   "boolean"
145 #define DTD_ENTITY_DEVPATH   "devpath"
146 #define DTD_ENTITY_DRIVER    "driver"
147 #define DTD_ENTITY_DRVMIN    "drv_min"
148 #define DTD_ENTITY_FALSE     "false"
149 #define DTD_ENTITY_INT       "int"
150 #define DTD_ENTITY_STRING    "string"
151 #define DTD_ENTITY_TRUE      "true"
152 #define DTD_ENTITY_UINT      "uint"

154 #define DTD_ENTITY_BOOL_LEN  6      /* "false" */

156 #define ATTACH_FORCED        "SUNWattached.xml"

158 #define TMP_POOL_NAME         "SUNWtmp_%s"
159 #define MAX_TMP_POOL_NAME     (ZONENAME_MAX + 9)
160 #define RCAP_SERVICE          "system/rcap:default"
161 #define POOLD_SERVICE         "system/pools/dynamic:default"

163 /*
164  * rctl alias definitions
165  *
166  * This holds the alias, the full rctl name, the default priv value, action
167  * and lower limit. The functions that handle rctl aliases step through
168  * this table, matching on the alias, and using the full values for setting
169  * the rctl entry as well the limit for validation.
170  */
171 static struct alias {
172     char *shortname;
173     char *realname;
174     char *priv;
175     char *action;
176     uint64_t low_limit;
177 } aliases[] = {
_____ unchanged portion omitted
2960 /*
2961  * This is the nftw call-back function used by zonecfg_dev_manifest. It is
2962  * responsible for calling the actual call-back.
2963  */
2964 /* ARGSUSED2 */
2965 static int
2966 zonecfg_devwalk_cb(const char *path, const struct stat *st, int f,
2967     struct FTW *ftw)
2968 {
2969     acl_t *acl;
2970     char *acl_txt = NULL;

2972     /* skip all but character and block devices */
2973     if (!S_ISBLK(st->st_mode) && !S_ISCHR(st->st_mode))

```

```

2974         return (0);

2976     if ((acl_get(path, ACL_NO_TRIVIAL, &acl) == 0) &&
2977         acl != NULL) {
2978         acl_txt = acl_totext(acl, ACL_NORESOLVE);
2979         acl_free(acl);
2980     }

2982     if (strlen(path) <= g_devwalk_skip_prefix)
2983         return (0);

2985     (void) g_devwalk_cb(path + g_devwalk_skip_prefix, st->st_uid,
2986         st->st_gid, st->st_mode & S_IAMB, acl_txt != NULL ? acl_txt : "",
2987         g_devwalk_cb(path + g_devwalk_skip_prefix, st->st_uid, st->st_gid,
2988             st->st_mode & S_IAMB, acl_txt != NULL ? acl_txt : "",
2989             g_devwalk_data);
2990     free(acl_txt);
2991     return (0);
_____ }
_____ unchanged portion omitted

```