

new/usr/src/lib/gss_mechs/mech_krb5/Makefile.com

1

18855 Sat Jan 31 23:14:47 2015

new/usr/src/lib/gss_mechs/mech_krb5/Makefile.com

872 remove duplicate daemon() definitions

```
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 # Copyright (c) 1999, 2010, Oracle and/or its affiliates. All rights reserved.
22 #
23 # This make file will build mech_krb5.so.1. This shared object
24 # contains all the functionality needed to support the Kerberos V5 GSS-API
25 # mechanism. No other Kerberos libraries are needed.
26 #
27 #
28 LIBRARY= mech_krb5.a
29 VERS = .1
30 #
31 FILEMODE= 755
32 #
33 # objects are listed by source directory
34 REL_PATH= ../
35 #
36 to_all: all
37 #
38 # crypto
39 CRYPTO = cksumtype_to_string.o \
40 coll_proof_cksum.o enctype_compare.o enctype_to_string.o \
41 keyed_checksum_types.o keyed_cksum.o \
42 make_random_key.o string_to_cksumtype.o \
43 string_to_enctype.o string_to_key.o valid_cksumtype.o \
44 valid_enctype.o pkcs11slot.o state.o pbkdf2.o old_api_glue.o \
45 keylengths.o random_to_key.o
46 #
47 CRYPTO_UTS= cksumtypes.o decrypt.o encrypt.o encrypt_length.o \
48 etypes.o nfold.o verify_checksum.o default_state.o \
49 prng.o block_size.o make_checksum.o checksum_length.o hmac.o \
50 mandatory_sumtype.o combine_keys.o
51 #
52 # crypto/raw
53 CRYPTO_RAW= raw_decrypt.o raw_encrypt.o
54 #
55 # crypto/des user space only
56 CRYPTO_DES= afsstring2key.o string2key.o
57 #
58 # crypto/des common to user and kernel space
59 CRYPTO_DES_UTS= f CBC.o f_cksum.o f_parity.o weak_key.o d3 CBC.o
60 #
61 # crypto/arcfour user space only
```

new/usr/src/lib/gss_mechs/mech_krb5/Makefile.com

2

```
62 CRYPTO_ARCFOUR= arcfour_str2key.o
63 #
64 # crypto/aes user space only
65 CRYPTO_AES= aes_s2k.o
66 #
67 # crypto/arcfour common to user and kernel space
68 CRYPTO_ARCFOUR_UTS = k5_arcfour.o
69 #
70 #crypto/dk
71 CRYPTO_DK= stringtokey.o
72 #
73 CRYPTO_DK_UTS= checksum.o derive.o dk_decrypt.o dk_encrypt.o
74 #
75 # crypto/crc32
76 CRYPTO_CRC32= crc.o
77 #
78 # crypto/crc32
79 CRYPTO_CRC32_UTS= crc32.o
80 #
81 # crypto/md4
82 CRYPTO_MD4= md4.o
83 #
84 # crypto/enc_provider
85 CRYPTO_ENC= des.o des3.o arcfour_provider.o aes_provider.o
86 #
87 # crypto/hash_provider
88 CRYPTO_HASH= hash_md5.o hash_shal.o hash_ef_generic.o
89 CRYPTO_HASH_UTS= hash_crc32.o
90 #
91 # crypto/keyhash_provider
92 CRYPTO_KEYHASH= k5_md5des.o hmac_md5.o
93 CRYPTO_KEYHASH_UTS= descbc.o
94 #
95 # crypto/old
96 CRYPTO_OLD= des_stringtokey.o
97 #
98 # crypto/old
99 CRYPTO_OLD_UTS= old_encrypt.o old_decrypt.o
100 #
101 # et error_tables
102 ET= adb_err.o adm_err.o asnl_err.o chpass_util_strings.o error_message.o \
103 com_err.o gssapi_err_generic.o import_err.o \
104 gssapi_err_krb5.o kadm_err.o kdb5_err.o kpasswd_strings.o kdc5_err.o \
105 krb5_err.o kv5m_err.o prof_err.o pty_err.o ss_err.o
106 #
107 # krb5/asn.1
108 K5_ASN1= asnl_decode.o asnl_k_decode.o asnl_encode.o \
109 asnl_get.o asnl_make.o \
110 asnlbuf.o krb5_decode.o krb5_encode.o \
111 asnl_k_encode.o asnl_misc.o ldap_key_seq.o
112 #
113 # krb5/ccache
114 K5_CC= cc_file.o cc_memory.o ccbase.o ccfn.o ccdefault.o ccdefops.o ser_cc.o cc
115 #
116 # krb5/keytab
117 K5_KT= ktadd.o ktbase.o ktdefault.o ktfr_entry.o \
118 kremove.o read_servi.o kt_file.o kt_srvtab.o ktfn.o kt_findrealm.o \
119 kt_solaris.o
120 #
121 K5_KRB= addr_comp.o addr_order.o addr_srch.o \
122 auth_con.o bld_pr_ext.o bld_princ.o chk_trans.o \
123 conv_princ.o copy_addrs.o copy_creds.o copy_data.o copy_tick.o \
124 cp_key_cnt.o decode_kdc.o decrypt_tk.o encode_kdc.o encrypt_tk.o \
125 free_rtree.o fwd_tgt.o gc_frm_kdc.o gc_via_tkt.o gen_segnum.o \
126 gen_subkey.o get_creds.o get_in_tkt.o kdc_rep_dc.o mk_cred.o \
127 mk_error.o mk_priv.o mk_rep.o mk_req.o mk_req_ext.o mk_safe.o \
```

```

128 pr_to_salt.o princ_comp.o privacy_allowed.o rd_cred.o \
129 rd_error.o rd_priv.o rd_rep.o rd_req.o rd_req_dec.o rd_safe.o \
130 recvauth.o send_tgs.o sendauth.o srv_rcache.o str_conv.o \
131 tgtname.o valid_times.o walk_rtree.o appdefault.o deltato.o \
132 enc_helper.o gic_keytab.o gic_opt.o gic_pwd.o preauth2.o \
133 preauth.o vfy_increds.o vic_opt.o set_realm.o krb5_libinit.o chpw.o \
134 init_keyblock.o init_allocated_keyblock.o get_set_keyblock.o kerrs.o \
135 getuid.o pac.o

137 K5_KRB_UTS= copy_athctr.o copy_auth.o copy_cksum.o copy_key.o \
138 copy_princ.o init_ctx.o kfree.o parse.o ser_actx.o \
139 ser_adata.o ser_addr.o ser_auth.o \
140 ser_cksum.o ser_ctx.o ser_key.o \
141 ser_princ.o serialize.o unparse.o

143 K5_OS= an_to_ln.o def_realm.o cdefname.o free_krbhs.o free_hstrl.o \
144 full_ipadr.o get_krbhst.o gen_port.o genaddrs.o gen_rname.o \
145 gmt_mktime.o hostaddr.o hst_realm.o krbfileio.o \
146 ktdefname.o kuserok.o mk_faddr.o localaddr.o locate_kdc.o lock_file.o \
147 net_read.o net_write.o osconfig.o port2ip.o promptusr.o \
148 read_msg.o read_pwd.o realm_dom.o sendto_kdc.o sn2princ.o \
149 unlck_file.o ustime.o write_msg.o safeshown.o \
150 prompter.o realm_iter.o foreachaddr.o \
151 dnsglue.o dnssrv.o thread_safe.o changepw.o accessor.o

153 K5_OS_UTS=init_os_ctx.o timeofday.o toffset.o c_ustime.o

155 K5_POSIX= setenv.o
155 K5_POSIX= setenv.o daemon.o

157 K5_RCACHE=rc_base.o rc_file.o rc_mem.o rc_common.o rc_io.o rcdef.o rc_conv.o \
158 ser_rc.o rcfns.o rc_none.o

160 MECH= accept_sec_context.o store_cred.o \
161 add_cred.o disp_com_err_status.o disp_major_status.o \
162 compare_name.o context_time.o copy_ccache.o \
163 disp_name.o disp_status.o export_sec_context.o \
164 get_tkt_flags.o import_name.o indicate_mechs.o \
165 inq_context.o inq_cred.o inq_names.o \
166 krb5_gss_glue.o \
167 pname_to_uid.o process_context_token.o \
168 rel_buffer.o rel_oid.o rel_oid_set.o \
169 rel_cred.o rel_name.o util_buffer.o \
170 util_dup.o util_localhost.o \
171 util_cksum.o acquire_cred.o init_sec_context.o \
172 set_ccache.o acquire_cred_with_pw.o lucid_context.o \
173 set_allowable_enctypes.o oid_ops.o export_name.o gss_libinit.o \
174 util_buffer_set.o util_errmap.o

176 MECH_UTS= delete_sec_context.o gssapi_krb5.o \
177 import_sec_context.o k5seal.o k5sealv3.o \
178 k5unseal.o seal.o ser_sctx.o \
179 sign.o unseal.o util_crypt.o \
180 util_ordering.o util_seed.o util_seqnum.o \
181 util_set.o util_token.o util_validate.o \
182 val_cred.o verify.o wrap_size_limit.o

184 GSSAPI_UTS= gen_oids.o

186 PROFILE_OBJS= prof_tree.o prof_file.o prof_parse.o prof_init.o \
187 prof_set.o prof_get.o prof_solaris.o

189 SUPPORT_OBJS= fake_addrinfo.o init_addrinfo.o threads.o errors.o plugins.o \
190 utf8_conv.o utf8.o

192 KWARN_OBJS= kwarnd_clnt_stubs.o kwarnd_clnt.o kwarnd_handle.o kwarnd_xdr.o

```

```

194 OBJECTS= \
195 $(MECH) $(MECH_UTS) $(GSSAPI_UTS)\
196 $(SUPPORT_OBJS) \
197 $(KWARN_OBJS) \
198 $(PROFILE_OBJS) \
199 $(CRYPTO) $(CRYPTO_UTS) \
200 $(CRYPTO_CRC32) \
201 $(CRYPTO_CRC32_UTS) \
202 $(CRYPTO_DES) $(CRYPTO_DES_UTS) \
203 $(CRYPTO_MD4) \
204 $(CRYPTO_DK) $(CRYPTO_DK_UTS) \
205 $(CRYPTO_ARCFOUR) $(CRYPTO_ARCFOUR_UTS) \
206 $(CRYPTO_AES) \
207 $(CRYPTO_ENC) \
208 $(CRYPTO_HASH) $(CRYPTO_HASH_UTS) \
209 $(CRYPTO_KEYHASH) $(CRYPTO_KEYHASH_UTS) \
210 $(CRYPTO_OLD) $(CRYPTO_OLD_UTS) \
211 $(CRYPTO_RAW) \
212 $(ET) \
213 $(K5_ASN1) \
214 $(K5_CC) \
215 $(K5_KT) \
216 $(K5_KRB) $(K5_KRB_UTS) \
217 $(K5_OS) $(K5_OS_UTS) \
218 $(K5_POSIX) $(K5_RCACHE)

220 # include library definitions
221 include $(REL_PATH)/../Makefile.lib

223 # Must come after Makefile.lib so CPPFLAGS doesn't get overwritten
224 include $(SRC)/lib/gss_mechs/mech_krb5/Makefile.mech_krb5

226 K5LIBLINK=$(LIBRARY:%.a=lib%.so)

228 # override default text domain
229 TEXT_DOMAIN= SUNW_OST_NETRPC
230 INS.liblink2= -$(RM) $@; $(SYMLINK) gss/$(LIBLINKPATH)$(LIBLINKS) $@

232 CPPFLAGS += -I$(REL_PATH)/libgss -I../include \
233 -I$(SRC)/uts/common/gssapi \
234 -I$(SRC)/uts/common/gssapi/include \
235 -I$(SRC)/lib/gss_mechs/mech_krb5/mech \
236 -I$(SRC)/lib/gss_mechs/mech_krb5/include/krb5 \
237 -I../include/krb5 \
238 -I../krb5/keytab \
239 -I../krb5/krb \
240 -I../krb5/os \
241 -I../krb5/ccache \
242 -I../krb5/rcache \
243 -I$(SRC)/lib/krb5 \
244 -I$(SRC)/lib/krb5/kadm5 \
245 -I$(SRC)/uts/common/gssapi/mechs/krb5/include \
246 -I$(SRC)/uts/common/gssapi/mechs/krb5/crypto/des

248 # KRB5_DEFS can be assigned various preprocessor flags, typically -D
249 # defines on the make invocation. These values will be appended to
250 # CPPFLAGS so the other flags in CPPFLAGS will not be overwritten.

252 CPPFLAGS += $(KRB5_DEFS)

254 CERRWARN += -_gcc=-Wno-unused-function
255 CERRWARN += -_gcc=-Wno-type-limits
256 CERRWARN += -_gcc=-Wno-uninitialized
257 CERRWARN += -_gcc=-Wno-parentheses
258 CERRWARN += -_gcc=-Wno-unused-variable

```

```

259 CERRWARN +=      _gcc=-Wno-unused-label
260 CERRWARN +=      _gcc=-Wno-unused-value
261 CERRWARN +=      _gcc=-Wno-empty-body
262 CERRWARN +=      _gcc=-Wno-address

264 MAPFILES =      ../mapfile-vers

266 #CPPFLAGS +=      -D_REENTRANT
267 $(PICS) :=        CFLAGS += $(XFFLAG)
268 $(PICS) :=        CFLAGS64 += $(XFFLAG)
269 $(PICS) :=        CCFLAGS += $(XFFLAG)
270 $(PICS) :=        CCFLAGS64 += $(XFFLAG)

272 LIBS = $(DYNLIB) $(K5LIBLINK)

274 # override ROOTLIBDIR and ROOTLINKS
275 ROOTLIBDIR=      $(ROOT)/usr/lib/gss
276 ROOTLIBDIR64=    $(ROOT)/usr/lib/$(MACH64)/gss

278 K5MECHLINK=      $(K5LIBLINK:%=$(ROOT)/usr/lib/%)
279 K5MECHLINK64=    $(K5LIBLINK:%=$(ROOT)/usr/lib/$(MACH64)/%)

281 ROOTLIBS=        $(LIBS:%=$(ROOTLIBDIR)/%) $(K5MECHLINK)
282 ROOTLIBS64=      $(LIBS:%=$(ROOTLIBDIR64)/%) $(K5MECHLINK64)

284 $(ROOTLIBDIR) $(ROOTLIBDIR64):
285     $(INS.dir)

287 # create libmech_krb5 link locally
288 $(K5LIBLINK): $(DYNLIB)
289     -$(RM) $@; $(SYMLINK) $(DYNLIB) $@

291 # create libmech_krb5 link in $ROOT/usr/lib/gss/
292 $(ROOTLIBDIR)/$(K5LIBLINK):      $(ROOTLIBDIR)/$(LIBLINKS)$(VERS)
293     $(INS.liblink)

295 # create libmech_krb5 link in $ROOT/usr/lib/$(MACH64)/gss
296 $(ROOTLIBDIR64)/$(K5LIBLINK):    $(ROOTLIBDIR64)/$(LIBLINKS)$(VERS)
297     $(INS.liblink)

299 # create libmech_krb5 link in ROOT/usr/lib
300 $(K5MECHLINK): $(ROOTLIBDIR)/$(LIBLINKS)$(VERS)
301     $(INS.liblink2)

303 # create libmech_krb5 link in ROOT/usr/lib/$(MACH64)
304 $(K5MECHLINK64):      $(ROOTLIBDIR64)/$(LIBLINKS)$(VERS)
305     $(INS.liblink2)

307 LDLIBS += -lgss -lsocket -lresolv -lc -lpkcs11 -lnsl -lkstat
308 # -z ignore causes linker to ignore unneeded dependencies. This is
309 # needed because -lnsl is only used if DEBUG is defined.
310 DYNFLAGS += $(ZIGNORE)

312 # mech lib needs special initialization at load time
313 DYNFLAGS += -zinitarray=krb5_ld_init

315 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/%.c
316     $(COMPILE.c) -o $@ $<
317     $(POST_PROCESS_O)

319 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/mech/%.c
320     $(COMPILE.c) -o $@ $<
321     $(POST_PROCESS_O)

323 objs/%.o pics/%.o: $(SRC)/lib/gss_mechs/mech_krb5/mech/%.c
324     $(COMPILE.c) -o $@ $<

```

```

325     $(POST_PROCESS_O)

327 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/%.c
328     $(COMPILE.c) -o $@ $<
329     $(POST_PROCESS_O)

331 objs/%.o pics/%.o: $(REL_PATH)/crypto/%.c
332     $(COMPILE.c) -o $@ $<
333     $(POST_PROCESS_O)

335 objs/%.o pics/%.o: $(REL_PATH)/crypto/md4/%.c
336     $(COMPILE.c) -o $@ $<
337     $(POST_PROCESS_O)

339 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/des/%.c
340     $(COMPILE.c) -o $@ $<
341     $(POST_PROCESS_O)

343 objs/%.o pics/%.o: $(REL_PATH)/crypto/des/%.c
344     $(COMPILE.c) -o $@ $<
345     $(POST_PROCESS_O)

347 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/arcfour/%.c
348     $(COMPILE.c) -o $@ $<
349     $(POST_PROCESS_O)

351 objs/%.o pics/%.o: $(REL_PATH)/crypto/arcfour/%.c
352     $(COMPILE.c) -o $@ $<
353     $(POST_PROCESS_O)

355 objs/%.o pics/%.o: $(REL_PATH)/crypto/aes/%.c
356     $(COMPILE.c) -o $@ $<
357     $(POST_PROCESS_O)

359 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/dk/%.c
360     $(COMPILE.c) -o $@ $<
361     $(POST_PROCESS_O)

363 objs/%.o pics/%.o: $(REL_PATH)/crypto/dk/%.c
364     $(COMPILE.c) -o $@ $<
365     $(POST_PROCESS_O)

367 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/raw/%.c
368     $(COMPILE.c) -o $@ $<
369     $(POST_PROCESS_O)

371 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/crc32/%.c
372     $(COMPILE.c) -o $@ $<
373     $(POST_PROCESS_O)

375 objs/%.o pics/%.o: $(REL_PATH)/crypto/crc32/%.c
376     $(COMPILE.c) -o $@ $<
377     $(POST_PROCESS_O)

379 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/shal/%.c
380     $(COMPILE.c) -o $@ $<
381     $(POST_PROCESS_O)

383 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/enc_provider/%.c
384     $(COMPILE.c) -o $@ $<
385     $(POST_PROCESS_O)

387 objs/%.o pics/%.o: $(REL_PATH)/crypto/hash_provider/%.c
388     $(COMPILE.c) -o $@ $<
389     $(POST_PROCESS_O)

```

```

391 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/hash_provider/%.c
392     $(COMPILE.c) -o $@ $<
393     $(POST_PROCESS_O)

395 objs/%.o pics/%.o: $(REL_PATH)/crypto/keyhash_provider/%.c
396     $(COMPILE.c) -o $@ $<
397     $(POST_PROCESS_O)

399 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/keyhash_provider/%
400     $(COMPILE.c) -o $@ $<
401     $(POST_PROCESS_O)

403 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/old/%.c
404     $(COMPILE.c) -o $@ $<
405     $(POST_PROCESS_O)

407 objs/%.o pics/%.o: $(REL_PATH)/crypto/old/%.c
408     $(COMPILE.c) -o $@ $<
409     $(POST_PROCESS_O)

411 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/crypto/os/%.c
412     $(COMPILE.c) -o $@ $<
413     $(POST_PROCESS_O)

415 objs/%.o pics/%.o: $(REL_PATH)/crypto/shal/%.c
416     $(COMPILE.c) -o $@ $<
417     $(POST_PROCESS_O)

419 objs/%.o pics/%.o: $(REL_PATH)/et/%.c
420     $(COMPILE.c) -o $@ $<
421     $(POST_PROCESS_O)

423 objs/%.o pics/%.o: $(REL_PATH)/krb5/asn.1/%.c
424     $(COMPILE.c) -o $@ $<
425     $(POST_PROCESS_O)

427 objs/%.o pics/%.o: $(REL_PATH)/krb5/ccache/%.c
428     $(COMPILE.c) -o $@ $<
429     $(POST_PROCESS_O)

431 objs/%.o pics/%.o: $(REL_PATH)/krb5/ccache/file/%.c
432     $(COMPILE.c) -o $@ $<
433     $(POST_PROCESS_O)

435 objs/%.o pics/%.o: $(REL_PATH)/krb5/ccache/stdio/%.c
436     $(COMPILE.c) -o $@ $<
437     $(POST_PROCESS_O)

439 objs/%.o pics/%.o: $(REL_PATH)/krb5/ccache/memory/%.c
440     $(COMPILE.c) -o $@ $<
441     $(POST_PROCESS_O)

443 objs/%.o pics/%.o: $(REL_PATH)/krb5/keytab/%.c
444     $(COMPILE.c) -o $@ $<
445     $(POST_PROCESS_O)

447 objs/%.o pics/%.o: $(REL_PATH)/krb5/keytab/file/%.c
448     $(COMPILE.c) -o $@ $<
449     $(POST_PROCESS_O)

451 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/krb5/krb/%.c
452     $(COMPILE.c) -o $@ $<
453     $(POST_PROCESS_O)

455 objs/%.o pics/%.o: $(REL_PATH)/krb5/krb/%.c
456     $(COMPILE.c) -o $@ $<

```

```

457     $(POST_PROCESS_O)

459 objs/%.o pics/%.o: $(SRC)/uts/common/gssapi/mechs/krb5/krb5/os/%.c
460     $(COMPILE.c) -o $@ $<
461     $(POST_PROCESS_O)

463 objs/%.o pics/%.o: $(REL_PATH)/krb5/os/%.c
464     $(COMPILE.c) -o $@ $<
465     $(POST_PROCESS_O)

467 objs/%.o pics/%.o: $(REL_PATH)/krb5/posix/%.c
468     $(COMPILE.c) -o $@ $<
469     $(POST_PROCESS_O)

471 objs/%.o pics/%.o: $(REL_PATH)/krb5/rcache/%.c
472     $(COMPILE.c) -o $@ $<
473     $(POST_PROCESS_O)

475 objs/%.o pics/%.o: $(REL_PATH)/profile/%.c
476     $(COMPILE.c) -o $@ $<
477     $(POST_PROCESS_O)

479 objs/%.o pics/%.o: $(REL_PATH)/support/%.c
480     $(COMPILE.c) -o $@ $<
481     $(POST_PROCESS_O)

483 # include library targets
484 include $(REL_PATH)/../Makefile.targ
485 # We need to include all CPPFLAGS for the files since we are using
486 # the macro to expand and they are not aware of flags yet .. ugh..

488 OS_FLAGS = -DHAVE_LIBSOCKET -DHAVE_LIBNSL -DTIME_WITH_SYS_TIME \
489     -DHAVE_UNISTD_H -DHAVE_SYS_TIME_H -DHAVE_REGEX_H \
490     -DHAVE_REGEX_H -DHAVE_RE_COMP -DHAVE_REGCOMP \
491     -DPOSIX_TYPES -DNDBM \
492     -DHAVE_STDLIB_H -DHAVE_STDARG_H -DHAVE_SYS_TYPES_H \
493     -DHAVE_NETINET_IN_H -DHAVE_SRAND48 \
494     -DHAVE_SRAND -DHAVE_STRANDOM -DHAVE_GETPID \
495     -DHAVE_ERRNO -DHAVE_STRFTIME -DHAVE_STRPTIME -DHAVE_STRERROR \
496     -DHAVE_STAT -DSIZEOF_INT=4 -DPROVIDE_KERNEL_IMPORT \
497     -DHAVE_STDINT_H -DPOSIX_SIGNALS -DHAVE_GETENV -DHAVE_SETENV \
498     -DHAVE_UNSETENV -DHAVE_FCHMOD -DHAVE_STRUCT_LIFCONF \
499     -DHAVE_VASPRINTF -DHAVE_RES_NINIT -DHAVE_RES_NDESTROY \
500     -DHAVE_RES_NSEARCH

502 CPPFLAGS += -I$(REL_PATH)krb5/ccache/file $(OS_FLAGS)

504 SOURCES= \
505     $(CRYPTO_OS_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/os/%.c)\
506     $(K5_OS_UTS:%.o=$(SRC)/uts/common/gssapi/mechs/krb5/krb5/os/%.c) \
507     $(K5_OS:%.o=$(SRC)/lib/gss_mechs/mech_krb5/krb5/os/%.c) \
508     $(CRYPTO:%.o=$(SRC)/lib/gss_mechs/mech_krb5/crypto/%.c) \
509     $(CRYPTO_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/%.c)\
510     $(CRYPTO_RAW:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/raw/%.c)\
511     $(CRYPTO_DES:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/des/%.c)\
512     $(CRYPTO_DES_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/des/%.c)
513     $(CRYPTO_ARCFOUR:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/arcfour/%.c)
514     $(CRYPTO_ARCFOUR_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/arc
515     $(CRYPTO_AES:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/aes/%.c)\
516     $(CRYPTO_DK:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/dk/%.c)\
517     $(CRYPTO_DK_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/dk/%.c)\
518     $(CRYPTO_CRC32:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/crc32/%.c)\
519     $(CRYPTO_MD4:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/md4/%.c)\
520     $(CRYPTO_CRC32_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/crc32
521     $(CRYPTO_ENC:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/enc_provide
522     $(CRYPTO_HASH:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/hash_provider/%

```

```
523 $(CRYPTO_HASH_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/hash_p
524 $(CRYPTO_KEYHASH:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/keyhash_prov
525 $(CRYPTO_KEYHASH_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/key
526 $(CRYPTO_OLD:%.o= $(SRC)/lib/gss_mechs/mech_krb5/crypto/old/%.c)\
527 $(CRYPTO_OLD_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/crypto/old/%.c
528 $(ET:%.o= $(SRC)/lib/gss_mechs/mech_krb5/et/%.c) \
529 $(K5_ASN1:%.o= $(SRC)/lib/gss_mechs/mech_krb5/krb5/asn.1/%.c) \
530 $(K5_CC:%.o= $(SRC)/lib/gss_mechs/mech_krb5/krb5/ccache/%.c) \
531 $(K5_KT:%.o= $(SRC)/lib/gss_mechs/mech_krb5/krb5/keytab/%.c) \
532 $(K5_KRB:%.o= $(SRC)/lib/gss_mechs/mech_krb5/krb5/krb/%.c)\
533 $(K5_KRB_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/krb/%.c)\
534 $(K5_OS:%.o= $(SRC)/lib/gss_mechs/mech_krb5/krb5/os/%.c)\
535 $(K5_OS_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/krb5/os/%.c)\
536 $(K5_POSIX:%.o= $(SRC)/lib/gss_mechs/mech_krb5/krb5/posix/%.c)\
537 $(K5_RCACHE:%.o= $(SRC)/lib/gss_mechs/mech_krb5/krb5/rcache/%.c) \
538 $(MECH:%.o= $(SRC)/lib/gss_mechs/mech_krb5/mech/%.c) \
539 $(MECH_UTS:%.o= $(SRC)/uts/common/gssapi/mechs/krb5/mech/%.c) \
540 $(GSSAPI_UTS:%.o= $(SRC)/uts/common/gssapi/%.c) \
541 $(PROFILE_OBJS:%.o= $(SRC)/lib/gss_mechs/mech_krb5/profile/%.c) \
542 $(SUPPORT_OBJS:%.o= $(SRC)/lib/gss_mechs/mech_krb5/support/%.c)

544 kwarnd.h: $(SRC)/cmd/krb5/kwarn/kwarnd.x
545 $(RM) $@
546 $(RPCGEN) -M -h $(SRC)/cmd/krb5/kwarn/kwarnd.x | \
547 $(SED) -e 's!$(SRC)/cmd/krb5/kwarn/kwarnd.h!kwarnd.h!' > $@

549 kwarnd_xdr.c: kwarnd.h $(SRC)/cmd/krb5/kwarn/kwarnd.x
550 $(RM) $@
551 $(RPCGEN) -M -c $(SRC)/cmd/krb5/kwarn/kwarnd.x | \
552 $(SED) -e 's!$(SRC)/cmd/krb5/kwarn/kwarnd.h!kwarnd.h!' > $@

554 kwarnd_clnt.c: kwarnd.h $(SRC)/cmd/krb5/kwarn/kwarnd.x
555 $(RM) $@
556 $(RPCGEN) -M -l $(SRC)/cmd/krb5/kwarn/kwarnd.x | \
557 $(SED) -e 's!$(SRC)/cmd/krb5/kwarn/kwarnd.h!kwarnd.h!' > $@

559 kwarnd_clnt_stubs.c: kwarnd.h $(SRC)/cmd/krb5/kwarn/kwarnd_clnt_stubs.c
560 $(RM) $@
561 $(CP) $(SRC)/cmd/krb5/kwarn/kwarnd_clnt_stubs.c $@

563 kwarnd_handle.c: $(SRC)/cmd/krb5/kwarn/kwarnd_handle.c
564 $(RM) $@
565 $(CP) $(SRC)/cmd/krb5/kwarn/kwarnd_handle.c $@

567 CLOBBERFILES += kwarnd.h \
568 kwarnd_clnt.c kwarnd_clnt_stubs.c kwarnd_handle.c kwarnd_xdr.c

570 # So lint.out won't be needlessly recreated
571 lint: $(LINTOUT)

573 $(LINTOUT): $(SOURCES)
574 $(LINT.c) -o $(LIBNAME) $(SOURCES) > $(LINTOUT) 2>&1
```

19407 Sat Jan 31 23:14:47 2015

872 remove duplicate daemon() definitions

_____unchanged_portion_omitted_____

```

49 SYMBOL_VERSION SUNWprivate_1.1 {
50   global:
51     adb_error_table;
52     add_com_err_hook;
53     add_default_com_err_hook;
54     initialize_adb_error_table;
55     asnl2krb5_buf;
56     asnl_decode_addrtype;
57     asnl_decode_ap_options;
58     asnl_decode_authdata_elt;
59     asnl_decode_authdatatype;
60     asnl_decode_authorization_data;
61     asnl_decode_charstring;
62     asnl_decode_checksum;
63     asnl_decode_cksumtype;
64     asnl_decode_enc_kdc_rep_part;
65     asnl_decode_enc_sam_key;
66     asnl_decode_enc_sam_response_enc;
67     asnl_decode_encrypted_data;
68     asnl_decode_encryption_key;
69     asnl_decode_etype;
70     asnl_decode_etype_info;
71     asnl_decode_generalstring;
72     asnl_decode_generaltime;
73     asnl_decode_host_address;
74     asnl_decode_host_addresses;
75     asnl_decode_ia5string;
76     asnl_decode_int32;
77     asnl_decode_int;
78     asnl_decode_integer;
79     asnl_decode_kdc_options;
80     asnl_decode_kdc_rep;
81     asnl_decode_kdc_req;
82     asnl_decode_kdc_req_body;
83     asnl_decode_kerberos_time;
84     asnl_decode_krb5_flags;
85     asnl_decode_krb_cred_info;
86     asnl_decode_krb_safe_body;
87     asnl_decode_kvno;
88     asnl_decode_last_req;
89     asnl_decode_last_req_entry;
90     asnl_decode_msgtype;
91     asnl_decode_null;
92     asnl_decode_octet;
93     asnl_decode_octetstring;
94     asnl_decode_pa_data;
95     asnl_decode_passwdsequence;
96     asnl_decode_predicted_sam_response;
97     asnl_decode_principal_name;
98     asnl_decode_printablestring;
99     asnl_decode_realm;
100    asnl_decode_sam_challenge;
101    asnl_decode_sam_flags;
102    asnl_decode_sam_response;
103    asnl_decode_sequence_of_etype;
104    asnl_decode_sequence_of_krb_cred_info;
105    asnl_decode_sequence_of_pa_data;
106    asnl_decode_sequence_of_passwdsequence;
107    asnl_decode_sequence_of_ticket;

```

```

108     asnl_decode_ticket;
109     asnl_decode_ticket_flags;
110     asnl_decode_transited_encoding;
111     asnl_decode_ui_2;
112     asnl_decode_ui_4;
113     asnl_decode_unsigned_integer;
114     asnl_error_table;
115     asnl_get_sequence;
116     asnl_get_tag_2;
117     asnl_krb5_realm_copy;
118     asnl_make_etag;
119     asnl_make_id;
120     asnl_make_length;
121     asnl_make_sequence;
122     asnl_make_set;
123     asnl_make_string;
124     asnl_make_tag;
125     asnlbuf_create;
126     asnlbuf_destroy;
127     asnlbuf_ensure_space;
128     asnlbuf_expand;
129     asnlbuf_free;
130     asnlbuf_hex_unparse;
131     asnlbuf_imbed;
132     asnlbuf_insert_charstring;
133     asnlbuf_insert_octet;
134     asnlbuf_insert_octetstring;
135     asnlbuf_len;
136     asnlbuf_remains;
137     asnlbuf_remove_charstring;
138     asnlbuf_remove_octet;
139     asnlbuf_remove_octetstring;
140     asnlbuf_size;
141     asnlbuf_skiptail;
142     asnlbuf_sync;
143     asnlbuf_unparse;
144     asnlbuf_wrap_data;
145     com_err;
146     com_err_va;
147     daemon;
147     decode_krb5_alt_method;
148     decode_krb5_ap_rep;
149     decode_krb5_ap_rep_enc_part;
150     decode_krb5_ap_req;
151     decode_krb5_as_rep;
152     decode_krb5_as_req;
153     decode_krb5_authdata;
154     decode_krb5_authenticator;
155     decode_krb5_cred;
156     decode_krb5_enc_cred_part;
157     decode_krb5_enc_data;
158     decode_krb5_enc_kdc_rep_part;
159     decode_krb5_enc_priv_part;
160     decode_krb5_enc_sam_key;
161     decode_krb5_enc_sam_response_enc;
162     decode_krb5_enc_tkt_part;
163     decode_krb5_encryption_key;
164     decode_krb5_error;
165     decode_krb5_etype_info;
166     decode_krb5_kdc_req_body;
167     decode_krb5_pa_enc_ts;
168     decode_krb5_padata_sequence;
169     decode_krb5_predicted_sam_response;
170     decode_krb5_priv;
171     decode_krb5_pwd_data;
172     decode_krb5_pwd_sequence;

```

```

173 decode_krb5_safe;
174 decode_krb5_sam_challenge;
175 decode_krb5_sam_response;
176 decode_krb5_tgs_rep;
177 decode_krb5_tgs_req;
178 decode_krb5_ticket;
179 encode_krb5_alt_method;
180 encode_krb5_ap_rep;
181 encode_krb5_ap_rep_enc_part;
182 encode_krb5_ap_req;
183 encode_krb5_as_rep;
184 encode_krb5_as_req;
185 encode_krb5_authdata;
186 encode_krb5_authenticator;
187 encode_krb5_cred;
188 encode_krb5_enc_cred_part;
189 encode_krb5_enc_data;
190 encode_krb5_enc_kdc_rep_part;
191 encode_krb5_enc_priv_part;
192 encode_krb5_enc_sam_response_enc;
193 encode_krb5_enc_tkt_part;
194 encode_krb5_encryption_key;
195 encode_krb5_error;
196 encode_krb5_etype_info2;
197 encode_krb5_etype_info;
198 encode_krb5_kdc_req_body;
199 encode_krb5_pa_enc_ts;
200 encode_krb5_padata_sequence;
201 encode_krb5_predicted_sam_response;
202 encode_krb5_priv;
203 encode_krb5_pwd_data;
204 encode_krb5_pwd_sequence;
205 encode_krb5_safe;
206 encode_krb5_sam_challenge;
207 encode_krb5_sam_key;
208 encode_krb5_sam_response;
209 encode_krb5_tgs_rep;
210 encode_krb5_tgs_req;
211 encode_krb5_ticket;
212 error_message;
213 generic_gss_oid_to_str;
214 generic_gss_release_buffer;
215 gss_error_table;
216 gss_krb5int_get_tkt_flags;
217 gssint_g_display_major_status;
218 gssint_g_make_string_buffer;
219 gssint_mecherrmap_map;
220 gssint_mecherrmap_map_errcode;
221 gsskrb5_extract_authz_data_from_sec_context;
222 gss_krb5_ccache_name;
223 gss_krb5_copy_ccache;
224 gss_mech_initialize;
225 gss_mech_krb5;
226 gss_mech_krb5_old;
227 gss_mech_krb5_wrong;
228 gss_mech_set_krb5;
229 gss_mech_set_krb5_both;
230 gss_mech_set_krb5_old;
231 gss_nt_krb5_name;
232 gss_nt_krb5_principal;
233 gssspi_acquire_cred_with_password;
234 imp_error_table;
235 k5_ef_hash;
236 k5_ef_mac;
237 k5g_error_table;
238 kadm_error_table;

```

```

239 kdb5_error_table;
240 kdc5_error_table;
241 kg_confounder_size;
242 kg_ctx_externalize;
243 kg_ctx_internalize;
244 kg_ctx_size;
245 kg_decrypt;
246 kg_encrypt;
247 kg_encrypt_size;
248 kg_get_defcred;
249 kg_get_seq_num;
250 kg_make_confounder;
251 kg_make_seed;
252 kg_make_seq_num;
253 kg_seal;
254 kg_unseal;
255 kg_vdb;
256 kpwd_error_table;
257 krb5_425_conv_principal;
258 krb5_524_conv_principal;
259 krb5_address_compare;
260 krb5_address_order;
261 krb5_address_search;
262 krb5_aname_to_localname;
263 krb5_appdefault_boolean;
264 krb5_appdefault_string;
265 krb5_auth_con_free;
266 krb5_auth_con_genaddrs;
267 krb5_auth_con_get_checksum_func;
268 krb5_auth_con_getaddrs;
269 krb5_auth_con_getauthenticator;
270 krb5_auth_con_getflags;
271 krb5_auth_con_getivector;
272 krb5_auth_con_getkey;
273 krb5_auth_con_getlocalseqnumber;
274 krb5_auth_con_getlocalsubkey;
275 krb5_auth_con_getpermetypes;
276 krb5_auth_con_getrcache;
277 krb5_auth_con_getrecvsbkey;
278 krb5_auth_con_getremoteseqnumber;
279 krb5_auth_con_getremotesubkey;
280 krb5_auth_con_getsendsubkey;
281 krb5_auth_con_init;
282 krb5_auth_con_initivector;
283 krb5_auth_con_set_checksum_func;
284 krb5_auth_con_set_req_cksuatype;
285 krb5_auth_con_set_safe_cksuatype;
286 krb5_auth_con_setaddrs;
287 krb5_auth_con_setflags;
288 krb5_auth_con_setivector;
289 krb5_auth_con_setpermetypes;
290 krb5_auth_con_setports;
291 krb5_auth_con_setrcache;
292 krb5_auth_con_setrecvsbkey;
293 krb5_auth_con_setsendsubkey;
294 krb5_auth_con_setuseruserkey;
295 krb5_auth_to_rep;
296 krb5_build_principal;
297 krb5_build_principal_ext;
298 krb5_build_principal_va;
299 krb5_c_block_size;
300 krb5_c_checksum_length;
301 krb5_c_decrypt;
302 krb5_c_encrypt;
303 krb5_c_encrypt_length;
304 krb5_c_etype_compare;

```

```

305     krb5_c_free_state;
306     krb5_c_init_state;
307     krb5_c_is_coll_proof_cksum;
308     krb5_c_is_keyed_cksum;
309     krb5_c_keyed_checksum_types;
310     krb5_c_keylengths;
311     krb5_c_make_checksum;
312     krb5_c_make_random_key;
313     krb5_c_random_make_octets;
314     krb5_c_random_seed;
315     krb5_c_random_to_key;
316     krb5_c_string_to_key;
317     krb5_c_string_to_key_with_params;
318     krb5_c_valid_cksumtype;
319     krb5_c_valid_etype;
320     krb5_c_verify_checksum;
321     krb5_cc_close;
322     krb5_cc_copy_creds;
323     krb5_cc_default;
324     krb5_cc_default_name;
325     krb5_cc_destroy;
326     krb5_cc_dfl_ops;
327     krb5_cc_end_seq_get;
328     krb5_cc_file_ops;
329     krb5_cc_gen_new;
330     krb5_cc_get_name;
331     krb5_cc_get_principal;
332     krb5_cc_get_type;
333     krb5_cc_initialize;
334     krb5_cc_next_cred;
335     krb5_cc_register;
336     krb5_cc_remove_cred;
337     krb5_cc_resolve;
338     krb5_cc_retrieve_cred;
339     krb5_cc_set_default_name;
340     krb5_cc_set_flags;
341     krb5_cc_start_seq_get;
342     krb5_cc_store_cred;
343     krb5_change_cache;
344     krb5_change_password;
345     krb5_check_transited_list;
346     krb5_checksum_size;
347     krb5_cksumtype_to_string;
348     krb5_cksumtypes_length;
349     krb5_cksumtypes_list;
350     krb5_clear_error_message;
351     krb5_copy_addr;
352     krb5_copy_addresses;
353     krb5_copy_authdata;
354     krb5_copy_authenticator;
355     krb5_copy_checksum;
356     krb5_copy_creds;
357     krb5_copy_data;
358     krb5_copy_keyblock;
359     krb5_copy_keyblock_contents;
360     krb5_copy_keyblock_data;
361     krb5_copy_principal;
362     krb5_copy_ticket;
363     krb5_create_secure_file;
364     krb5_crypto_us_timeofday;
365     krb5_decode_authdata_container;
366     krb5_decode_kdc_rep;
367     krb5_decode_ticket;
368     krb5_decrypt_tkt_part;
369     krb5_default_pwd_prompt1;
370     krb5_default_pwd_prompt2;

```

```

371     krb5_defkeyname;
372     krb5_do_preauth;
373     krb5_encode_authdata_container;
374     krb5_encode_kdc_rep;
375     krb5_encrypt_helper;
376     krb5_encrypt_tkt_part;
377     krb5_etype_to_string;
378     krb5_etype_to_istring;
379     krb5_etypes_length;
380     krb5_etypes_list;
381     krb5_error_table;
382     krb5_externalize_data;
383     krb5_externalize_opaque;
384     krb5_fcc_ops;
385     krb5_find_serializer;
386     krb5_free_address;
387     krb5_free_addresses;
388     krb5_free_ap_rep;
389     krb5_free_ap_rep_enc_part;
390     krb5_free_ap_req;
391     krb5_free_authdata;
392     krb5_free_authenticator;
393     krb5_free_authenticator_contents;
394     krb5_free_checksum;
395     krb5_free_checksum_contents;
396     krb5_free_cksumtypes;
397     krb5_free_config_files;
398     krb5_free_context;
399     krb5_free_cred;
400     krb5_free_cred_contents;
401     krb5_free_cred_enc_part;
402     krb5_free_creds;
403     krb5_free_data;
404     krb5_free_data_contents;
405     krb5_free_default_realm;
406     krb5_free_ef_handle;
407     krb5_free_enc_kdc_rep_part;
408     krb5_free_enc_sam_response_enc;
409     krb5_free_enc_sam_response_enc_contents;
410     krb5_free_enc_tkt_part;
411     krb5_free_error;
412     krb5_free_error_message;
413     krb5_free_etype_info;
414     krb5_free_host_realm;
415     krb5_free_kdc_rep;
416     krb5_free_kdc_req;
417     krb5_free_keyblock;
418     krb5_free_keyblock_contents;
419     krb5_free_keytab_entry_contents;
420     krb5_free_krbhst;
421     krb5_free_ktypes;
422     krb5_free_last_req;
423     krb5_free_pa_data;
424     krb5_free_pa_enc_ts;
425     krb5_free_predicted_sam_response;
426     krb5_free_predicted_sam_response_contents;
427     krb5_free_principal;
428     krb5_free_priv;
429     krb5_free_priv_enc_part;
430     krb5_free_pwd_data;
431     krb5_free_pwd_sequences;
432     krb5_free_realm_string;
433     krb5_free_realm_tree;
434     krb5_free_safe;
435     krb5_free_sam_challenge;
436     krb5_free_sam_challenge_contents;

```



```

437 krb5_free_sam_response;
438 krb5_free_sam_response_contents;
439 krb5_free_tgt_creds;
440 krb5_free_ticket;
441 krb5_free_tickets;
442 krb5_free_tkt_authent;
443 krb5_free_uio;
444 krb5_free_unparsed_name;
445 krb5_fwd_tgt_creds;
446 krb5_gen_portaddr;
447 krb5_gen_replay_name;
448 krb5_generate_seq_number;
449 krb5_generate_subkey;
450 krb5_get_cred_from_kdc;
451 krb5_get_cred_from_kdc_renew;
452 krb5_get_cred_from_kdc_validate;
453 krb5_get_cred_via_tkt;
454 krb5_get_credentials;
455 krb5_get_credentials_renew;
456 krb5_get_credentials_validate;
457 krb5_get_default_config_files;
458 krb5_get_default_in_tkt_ktypes;
459 krb5_get_default_realm;
460 krb5_get_error_message;
461 krb5_get_host_realm;
462 krb5_get_init_creds;
463 krb5_get_init_creds_keytab;
464 krb5_get_init_creds_opt_alloc;
465 krb5_get_init_creds_opt_free;
466 krb5_get_init_creds_opt_init;
467 krb5_get_init_creds_opt_set_address_list;
468 krb5_get_init_creds_opt_set_etype_list;
469 krb5_get_init_creds_opt_set_forwardable;
470 krb5_get_init_creds_opt_set_pa;
471 krb5_get_init_creds_opt_set_preauth_list;
472 krb5_get_init_creds_opt_set_proxiable;
473 krb5_get_init_creds_opt_set_renew_life;
474 krb5_get_init_creds_opt_set_salt;
475 krb5_get_init_creds_opt_set_tkt_life;
476 krb5_get_init_creds_password;
477 __krb5_get_init_creds_password;
478 krb5_get_key_data;
479 krb5_get_key_etype;
480 krb5_get_key_length;
481 krb5_get_krbhst;
482 krb5_get_notification_message;
483 krb5_get_permitted_etypes;
484 krb5_get_profile;
485 krb5_get_prompt_types;
486 krb5_get_realm_domain;
487 krb5_get_renewed_creds;
488 krb5_get_server_rcache;
489 krb5_get_servername;
490 krb5_get_tgs_ktypes;
491 krb5_get_time_offsets;
492 krb5_get_validated_creds;
493 krb5_getenv;
494 krb5_gss_display_status2;
495 krb5_gss_import_name;
496 krb5_gss_oid_array;
497 krb5_gss_userok;
498 krb5_gss_register_acceptor_identity;
499 krb5_hmac;
500 krb5_init_allocated_keyblock;
501 krb5_init_context;
502 krb5_init_ef_handle;

```

```

503 krb5_init_keyblock;
504 krb5_init_secure_context;
505 krb5_internalize_opaque;
506 krb5_is_permitted_etype;
507 krb5_is_referral_realm;
508 krb5_is_thread_safe;
509 krb5_kdc_rep_decrypt_proc;
510 krb5_kt_add_ad_entries;
511 krb5_kt_add_entry;
512 krb5_kt_close;
513 krb5_kt_default;
514 krb5_kt_default_name;
515 krb5_kt_dfl_ops;
516 krb5_kt_end_seq_get;
517 krb5_kt_find_realm;
518 krb5_kt_free_entry;
519 krb5_kt_get_entry;
520 krb5_kt_get_name;
521 krb5_kt_get_type;
522 krb5_kt_next_entry;
523 krb5_kt_read_service_key;
524 krb5_kt_register;
525 krb5_kt_remove_by_realm;
526 krb5_kt_remove_by_svcprinc;
527 krb5_kt_remove_entry;
528 krb5_kt_resolve;
529 krb5_kt_start_seq_get;
530 krb5_kt_ad_validate;
531 krb5_ktf_ops;
532 krb5_ktf_writable_ops;
533 krb5_ktfile_add;
534 krb5_ktfile_close;
535 krb5_ktfile_end_get;
536 krb5_ktfile_get_entry;
537 krb5_ktfile_get_name;
538 krb5_ktfile_get_next;
539 krb5_ktfile_remove;
540 krb5_ktfile_resolve;
541 krb5_ktfile_ser_entry;
542 krb5_ktfile_start_seq_get;
543 krb5_ktfile_wresolve;
544 krb5_ktfileint_close;
545 krb5_ktfileint_delete_entry;
546 krb5_ktfileint_find_slot;
547 krb5_ktfileint_internal_read_entry;
548 krb5_ktfileint_openr;
549 krb5_ktfileint_openw;
550 krb5_ktfileint_read_entry;
551 krb5_ktfileint_size_entry;
552 krb5_ktfileint_write_entry;
553 krb5_kuserok;
554 krb5_libdefault_boolean;
555 krb5_locate_kdc;
556 krb5_lock_file;
557 krb5_make_full_ipaddr;
558 krb5_make_fulladdr;
559 krb5_max_dgram_size;
560 krb5_max_skdc_timeout;
561 krb5_mk_lcred;
562 krb5_mk_error;
563 krb5_mk_ncred;
564 krb5_mk_priv;
565 krb5_mk_rep;
566 krb5_mk_req;
567 krb5_mk_req_extended;
568 krb5_mk_safe;

```

```

569     krb5_net_read;
570     krb5_net_write;
571     krb5_nfold;
572     krb5_old_decrypt;
573     krb5_old_encrypt;
574     krb5_old_encrypt_length;
575     krb5_os_free_context;
576     krb5_os_get_tty_uio;
577     krb5_os_hostaddr;
578     krb5_os_init_context;
579     krb5_os_localaddr;
580     krb5_overridekeyname;
581     krb5_pac_add_buffer;
582     krb5_pac_free;
583     krb5_pac_get_buffer;
584     krb5_pac_get_types;
585     krb5_pac_init;
586     krb5_pac_parse;
587     krb5_pac_verify;
588     krb5_parse_name;
589     krb5_pname_to_uid;
590     krb5_principal2salt;
591     krb5_principal2salt_norealm;
592     krb5_principal_compare;
593     krb5_privacy_allowed;
594     krb5_process_padata;
595     krb5_prompter_posix;
596     krb5_raw_decrypt;
597     krb5_raw_encrypt;
598     krb5_raw_encrypt_length;
599     krb5_rc_close;
600     krb5_rc_default;
601     krb5_rc_default_name;
602     krb5_rc_file_close;
603     krb5_rc_file_close_no_free;
604     krb5_rc_file_destroy;
605     krb5_rc_file_expunge;
606     krb5_rc_file_get_name;
607     krb5_rc_file_get_span;
608     krb5_rc_file_init;
609     krb5_rc_file_ops;
610     krb5_rc_file_recover;
611     krb5_rc_file_resolve;
612     krb5_rc_file_store;
613     krb5_rc_free_entry;
614     krb5_rc_get_type;
615     krb5_rc_io_close;
616     krb5_rc_io_creat;
617     krb5_rc_io_destroy;
618     krb5_rc_io_mark;
619     krb5_rc_io_move;
620     krb5_rc_io_open;
621     krb5_rc_io_read;
622     krb5_rc_io_size;
623     krb5_rc_io_sync;
624     krb5_rc_io_unmark;
625     krb5_rc_io_write;
626     krb5_rc_mem_close;
627     krb5_rc_mem_destroy;
628     krb5_rc_mem_get_name;
629     krb5_rc_mem_get_span;
630     krb5_rc_mem_init;
631     krb5_rc_mem_ops;
632     krb5_rc_mem_recover;
633     krb5_rc_mem_resolve;
634     krb5_rc_mem_store;

```

```

635     krb5_rc_register_type;
636     krb5_rc_resolve;
637     krb5_rc_resolve_full;
638     krb5_rc_store;
639     krb5_rd_cred;
640     krb5_rd_error;
641     krb5_rd_priv;
642     krb5_rd_rep;
643     krb5_rd_req;
644     krb5_rd_req_decoded;
645     krb5_rd_req_decoded_anyflag;
646     krb5_rd_safe;
647     krb5_read_message;
648     krb5_read_password;
649     krb5_realm_compare;
650     krb5_realm_iterator;
651     krb5_realm_iterator_create;
652     krb5_realm_iterator_free;
653     krb5_recvauth;
654     krb5_recvauth_version;
655     krb5_register_serializer;
656     krb5_saltpype_to_string;
657     krb5_secure_config_files;
658     krb5_send_tgs;
659     krb5_sendauth;
660     krb5_sendto_kdc;
661     krb5_ser_address_init;
662     krb5_ser_auth_context_init;
663     krb5_ser_authdata_init;
664     krb5_ser_authenticator_init;
665     krb5_ser_ccache_init;
666     krb5_ser_checksum_init;
667     krb5_ser_context_init;
668     krb5_ser_keyblock_init;
669     krb5_ser_keytab_init;
670     krb5_ser_pack_bytes;
671     krb5_ser_pack_int32;
672     krb5_ser_principal_init;
673     krb5_ser_rcache_init;
674     krb5_ser_unpack_bytes;
675     krb5_ser_unpack_int32;
676     krb5_set_config_files;
677     krb5_set_debugging_time;
678     krb5_set_default_in_tkt_ktypes;
679     krb5_set_default_realm;
680     krb5_set_default_tgs_enctypes;
681     krb5_set_default_tgs_ktypes;
682     krb5_set_error_message;
683     krb5_set_key_data;
684     krb5_set_key_enctype;
685     krb5_set_key_length;
686     krb5_set_password;
687     krb5_set_password_using_ccache;
688     krb5_set_principal_realm;
689     krb5_set_real_time;
690     krb5_set_time_offsets;
691     krb5_setenv;
692     krb5_size_opaque;
693     krb5_skdc_timeout_l;
694     krb5_skdc_timeout_shift;
695     krb5_sname_to_principal;
696     krb5_string_to_cksuptype;
697     krb5_string_to_deltat;
698     krb5_string_to_enctype;
699     krb5_string_to_key;
700     krb5_string_to_saltpype;

```

```

701     krb5_string_to_timestamp;
702     krb5_sync_disk_file;
703     krb5_tgtname;
704     krb5_timeofday;
705     krb5_timestamp_to_sfstring;
706     krb5_timestamp_to_string;
707     krb5_unlock_file;
708     krb5_unpack_full_ipaddr;
709     krb5_unparse_name;
710     krb5_unparse_name_ext;
711     krb5_unsetenv;
712     krb5_us_timeofday;
713     krb5_use_enctype;
714     krb5_use_natural_time;
715     krb5_validate_times;
716     krb5_verify_init_creds;
717     krb5_verify_init_creds_opt_init;
718     krb5_verify_init_creds_opt_set_ap_req_nofail;
719     krb5_walk_realm_tree;
720     krb5_write_message;
721     krb5int_accessor;
722     krb5int_aes_encrypt;
723     krb5int_clear_error;
724     krb5int_close_plugin_dirs;
725     krb5int_cm_call_select;
726     krb5int_fini_fac;
727     krb5int_foreach_localaddr;
728     krb5int_free_error;
729     krb5int_free_plugin_dir_data;
730     krb5int_gai_strerror;
731     krb5int_get_error;
732     krb5int_getnameinfo;
733     krb5int_get_plugin_dir_data;
734     krb5int_getspecific;
735     krb5int_gmt_mktime;
736     krb5int_hash_shal;
737     krb5int_init_context_kdc;
738     krb5int_init_fac;
739     krb5int_mutex_alloc;
740     krb5int_mutex_free;
741     krb5int_open_plugin_dirs;
742     krb5int_pbkdf2_hmac_shal;
743     krb5int_pthread_loaded;
744     krb5int_key_register;
745     krb5int_sendtokdc_debug_handler;
746     krb5int_setspecific;
747     krb5int_vset_error;
748     kwarn_add_warning;
749     kwarn_del_warning;
750     kv5m_error_table;
751     mit_afs_string_to_key;
752     mit_des_check_key_parity;
753     mit_des_fixup_key_parity;
754     mit_des_is_weak_key;
755     mit_des_string_to_key_int;
756     ovk_error_table;
757     ovku_error_table;
758     prof_error_table;
759     profile_add_node;
760     profile_add_relation;
761     profile_clear_relation;
762     profile_close_file;
763     profile_create_node;
764     profile_find_node_relation;
765     profile_find_node_subsection;
766     profile_free_node;

```

```

767     profile_free_list;
768     profile_get_integer;
769     profile_get_node_parent;
770     profile_get_options_boolean;
771     profile_get_options_string;
772     profile_get_string;
773     profile_get_values;
774     profile_init;
775     profile_init_path;
776     profile_open_file;
777     profile_parse_file;
778     profile_release;
779     profile_release_string;
780     profile_ser_externalize;
781     profile_ser_internalize;
782     profile_ser_size;
783     profile_verify_node;
784     __profile_abandon;
785     __profile_add_domain_mapping;
786     __profile_add_realm;
787     __profile_add_realm_entry;
788     __profile_add_xrealm_mapping;
789     __profile_get_default_realm;
790     __profile_get_realm_entry;
791     __profile_get_realms;
792     __profile_init;
793     __profile_release;
794     __profile_remove_domain_mapping;
795     __profile_remove_realm;
796     __profile_remove_xrealm_mapping;
797     __profile_set_appdefaults;
798     __profile_set_kdc;
799     __profile_set_libdefaults;
800     __profile_set_logging;
801     __profile_validate;
802     pty_error_table;
803     rem_com_err_hook;
804     rem_default_com_err_hook;
805     reset_com_err_hook;
806     set_com_err_hook;
807     ss_error_table;
808     local:
809     *;
810 };

```

_____ unchanged portion omitted

```

*****
4508 Sat Jan 31 23:14:47 2015
new/usr/src/lib/libresolv2/Makefile.com
872 remove duplicate daemon() definitions
*****
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 2009 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
25 #
27 LIBRARY= libresolv.a
28 VERS= .2

30 BSDOBJS=      putenv.o      strcasecmp.o   strsep.o \
30 BSDOBJS=      daemon.o      putenv.o       strcasecmp.o   strsep.o \
31              ftruncate.o   readv.o       strdup.o       strtoul.o \
32              gettimeofday.o setenv.o       strerror.o     utimes.o \
33              mktemp.o      setitimer.o   strpbrk.o     writev.o

35 DSTOBSJ=     dst_api.o      support.o      hmac_link.o

37 # inet_addr, inet_pton, inet_ntop, and inet_ntoa removed due to overlap with
38 # libnsl
39 INETOBJS=     inet_net_pton.o inet_neta.o    inet_lnaof.o \
40              inet_netof.o  nsap_addr.o   inet_makeaddr.o \
41              inet_network.o inet_net_ntop.o inet_cidr_ntop.o \
42              inet_cidr_pton.o inet_data.o

44 # build only the IRS objects that the ISC libbind's make would
45 IRSTHROBJS=   gethostent_r.o getnetent_r.o getnetgrent_r.o \
46              getprotoent_r.o getservent_r.o
47 IRSOBJS=     ${IRSTHROBJS} \
48              dns.o      dns_ho.o      dns_nw.o      dns_pr.o \
49              dns_sv.o   gai_strerror.o gen.o         gen_ho.o \
50              gen_ng.o   gen_nw.o     gen_pr.o     gen_sv.o \
51              getaddrinfo.o gethostent.o getnameinfo.o getnetent.o \
52              getnetgrent.o getprotoent.o getservent.o hesiod.o \
53              irp.o      irp_ho.o     irp_ng.o     irp_nw.o \
54              irp_pr.o   irp_sv.o     irpmarshall.o irs_data.o \
55              lcl.o      lcl_ho.o     lcl_ng.o     lcl_nw.o \
56              lcl_pr.o   lcl_sv.o     nis.o        nul_ng.o \
57              util.o

59 ISCOBJS=     assertions.o base64.o      bitncmp.o     ctl_clnt.o \
60              ctl_p.o   ctl_srvr.o   ev_connects.o ev_files.o \

```

```

61          ev_streams.o   ev_timers.o   ev_waits.o    eventlib.o \
62          heap.o         hex.o         logging.o      memcluster.o \
63          movefile.o     tree.o

65 NAMESEROBJS= ns_date.o     ns_name.o     ns_netint.o    ns_parse.o \
66              ns_print.o  ns_samedomain.o ns_sign.o     ns_ttl.o \
67              ns_verify.o ns_rdata.o    ns_newmsg.o

69 RESOLVOBJS=   herror.o      mtctxres.o    res_comp.o     res_data.o \
70              res_debug.o res_findzonecut.o res_init.o \
71              res_mkquery.o res_mkupdate.o res_query.o    res_send.o \
72              res_sendsigned.o res_update.o

74 SUNWOBJ=      sunw_mtctxres.o sunw_updrec.o sunw_wrappers.o

76 OBJECTS=      $(BSDOBJS) $(DSTOBSJ) $(INETOBJS) $(IRSOBJS) $(ISCOBJS) \
77              $(NAMESEROBJS) $(RESOLVOBJS) $(SUNWOBJ)

79 # include library definitions
80 include ../../Makefile.lib

82 # install this library in the root filesystem
83 include ../../Makefile.rootfs

85 # CC -v complains about things we aren't going to change in the ISC code
86 CCVERBOSE=

88 SRCDIR =      ../../common
89 SRCS=         $(BSDOBJS:%o=../../common/bsd/%c) \
90              $(DSTOBSJ:%o=../../common/dst/%c) \
91              $(INETOBJS:%o=../../common/inet/%c) \
92              $(IRSOBJS:%o=../../common/irs/%c) \
93              $(ISCOBJS:%o=../../common/isc/%c) \
94              $(NAMESEROBJS:%o=../../common/nameser/%c) \
95              $(RESOLVOBJS:%o=../../common/resolv/%c) \
96              $(SUNWOBJ:%o=../../common/sunw/%c)

98 LIBS =        $(DYNLIB) $(LINTLIB)

100 $(LINTLIB):= SRCS = ../../common/llib-lresolv

102 # Local Libresolv definitions

104 SOLCOMPAT =   -Dsocket=_socket
105 CRYPTFLAGS=   -DHMAC_MD5 -DUSE_MD5

107 LOCLFLAGS +=  $(CRYPTFLAGS)
108 LOCLFLAGS +=  -D_SYS_STREAM_H -D_REENTRANT -DSVR4 -DSUNW_OPTIONS \
109              $(SOLCOMPAT) -I../include -I../common/inc

111 CPPFLAGS +=   $(LOCLFLAGS)

113 CERRWARN +=   -gcc=-Wno-implicit-function-declaration

115 DYNFLGS +=    $(ZNODELETE)

117 LDLIBS +=     -lsocket -lnsl -lc -lmd

119 .KEEP_STATE:

121 all:          $(LIBS)

123 lint:         lintcheck

125 # include library targets
126 include ../../Makefile.targ

```

```
128 pics/%.o: ../common/bsd/%.c
129     $(COMPILE.c) -o $@ $<
130     $(POST_PROCESS_O)

132 pics/%.o: ../common/dst/%.c
133     $(COMPILE.c) -o $@ $<
134     $(POST_PROCESS_O)

136 pics/%.o: ../common/inet/%.c
137     $(COMPILE.c) -o $@ $<
138     $(POST_PROCESS_O)

140 pics/%.o: ../common/irs/%.c
141     $(COMPILE.c) -o $@ $<
142     $(POST_PROCESS_O)

144 pics/%.o: ../common/isc/%.c
145     $(COMPILE.c) -o $@ $<
146     $(POST_PROCESS_O)

148 pics/%.o: ../common/nameser/%.c
149     $(COMPILE.c) -o $@ $<
150     $(POST_PROCESS_O)

152 pics/%.o: ../common/resolv/%.c
153     $(COMPILE.c) -o $@ $<
154     $(POST_PROCESS_O)

156 pics/%.o: ../common/sunw/%.c
157     $(COMPILE.c) -o $@ $<
158     $(POST_PROCESS_O)

160 # install rule for lint library target
161 $(ROOTLINTDIR)/%: ../common/%
162     $(INS.file)
```

```

*****
6681 Sat Jan 31 23:14:47 2015
new/usr/src/lib/libresolv2/common/mapfile-vers
872 remove duplicate daemon() definitions
*****
_____unchanged_portion_omitted_____

116 SYMBOL_VERSION SUNWprivate_2.1 {
117     global:
118         __assertion_failed;
119         assertion_type_to_text;
120         b64_ntop;
121         __b64_ntop;
122         b64_pton;
123         __b64_pton;
124         __ctl_client;
125         __ctl_command;
126         __ctl_endclient;
127         __ctl_endserver;
128         __ctl_getcsctx;
129         __ctl_response;
130         __ctl_sendhelp;
131         __ctl_server;
132         __ctl_setcsctx;
133     daemon;
133     dn_count_labels;
134     __dn_count_labels;
135     dn_skipname;
136     __dst_buffer_to_key;
137     __dst_check_algorithm;
138     __dst_dnskey_to_key;
139     __dst_free_key;
140     __dst_generate_key;
141     __dst_init;
142     __dst_key_to_buffer;
143     __dst_read_key;
144     __dst_s_calculate_bits;
145     __dst_s_dns_key_id;
146     __dst_s_get_intl6;
147     __dst_s_id_calc;
148     __dst_sign_data;
149     __dst_s_put_intl6;
150     __dst_s_verify_str;
151     dst_t_func;
152     __dst_verify_data;
153     __dst_write_key;
154     __evAddTime;
155     __evCancelConn;
156     __evCancelRW;
157     __evClearTimer;
158     __evConnect;
159     __evConsIovec;
160     __evConsTime;
161     __evCreate;
162     __evDeselectFD;
163     __evDestroy;
164     __evDispatch;
165     __evDo;
166     __evGetNext;
167     __evHighestFD;
168     __evListen;
169     __evNowTime;
170     __evRead;
171     __evResetTimer;
172     __evSelectFD;
173     __evSetDebug;

```

```

174     __evSetTimer;
175     __evTimeVal;
176     __evUnwait;
177     __evWaitFor;
178     fp_nquery;
179     __fp_nquery;
180     fp_query;
181     __fp_resstat;
182     h_errlist;
183     __h_errno;
184     __h_errno_set;
185     hostalias;
186     inet_cidr_ntop;
187     inet_cidr_pton;
188     inet_neta;
189     inet_net_ntop;
190     inet_net_pton;
191     inet_nsap_addr;
192     inet_nsap_ntoa;
193     __irs_destroy;
194     __ISC_res_freeupdrec;
195     __ISC_res_mkupdate;
196     __ISC_res_mkupdrec;
197     __ISC_res_nmupdate;
198     __ISC_res_nupdate;
199     __ISC_res_update;
200     key_file_fmt_str;
201     loc_aton;
202     __loc_aton;
203     loc_ntoa;
204     __loc_ntoa;
205     __log_add_channel;
206     __log_category_is_active;
207     __log_check;
208     __log_close_stream;
209     __log_free_channel;
210     __log_free_context;
211     __log_get_channel_type;
212     __log_get_stream;
213     __log_inc_references;
214     __log_new_context;
215     __log_new_file_channel;
216     __log_new_null_channel;
217     __log_new_syslog_channel;
218     __log_open_stream;
219     __log_option;
220     __log_set_file_owner;
221     __log_vwrite;
222     __log_write;
223     __memget;
224     __memget_record;
225     __mempup;
226     __mempup_record;
227     __memstats;
228     __net_data_init;
229     __net_data_minimize;
230     __ns_datetosecs;
231     __ns_find_tsig;
232     __ns_flagdata;
233     __ns_format_ttl;
234     __ns_getl6;
235     __ns_get32;
236     __ns_initparse;
237     __ns_makecanon;
238     __ns_msg_getflag;
239     __ns_name_compress;

```

```

240     __ns_name_eq;
241     __ns_name_labels;
242     __ns_name_length;
243     __ns_name_map;
244     __ns_name_ntol;
245     __ns_name_ntop;
246     __ns_name_owned;
247     __ns_name_pack;
248     __ns_name_pton;
249     __ns_name_rollback;
250     __ns_name_skip;
251     __ns_name_uncompress;
252     __ns_name_unpack;
253     __ns_newmsg_copy;
254     __ns_newmsg_done;
255     __ns_newmsg_flag;
256     __ns_newmsg_id;
257     __ns_newmsg_init;
258     __ns_newmsg_q;
259     __ns_newmsg_rr;
260     __ns_parserr;
261     __ns_parse_ttl;
262     __ns_parserr2;
263     __ns_put16;
264     __ns_put32;
265     __ns_rdata_equal;
266     __ns_rdata_refers;
267     __ns_rdata_unpack;
268     __ns_samedomain;
269     __ns_samename;
270     __ns_sign;
271     __ns_sign2;
272     __ns_sign_tcp;
273     __ns_sign_tcp2;
274     __ns_sign_tcp_init;
275     __ns_skiprr;
276     __ns_sprintrr;
277     __ns_sprintrrf;
278     __ns_verify;
279     __ns_verify_tcp;
280     __ns_verify_tcp_init;
281     p_cdname;
282     __p_cert_syms;
283     p_class;
284     __p_class_syms;
285     p_fqname;
286     __p_key_syms;
287     p_option;
288     __p_option;
289     p_query;
290     p_rcode;
291     p_secstodate;
292     __p_secstodate;
293     p_section;
294     __p_section;
295     p_time;
296     p_type;
297     __p_type_syms;
298     putlong;
299     putshort;
300     __putshort;
301     res_close;
302     __res_disable_mt;
303     res_dnok;
304     __res_dnok;
305     __res_enable_mt;

```

```

306     res_endhostent;
307     res_findzonecut;
308     res_freehostent;
309     res_freeupdrec;
310     __res_get_h_errno;
311     res_gethostbyaddr;
312     res_gethostbyname;
313     res_gethostent;
314     res_getipnodebyaddr;
315     res_getipnodebyname;
316     __res_get_res;
317     res_hnok;
318     __res_hnok;
319     res_isourserver;
320     res_mailok;
321     __res_mailok;
322     res_mkupdate;
323     res_mkupdrec;
324     res_nameinquery;
325     __res_nameinquery;
326     res_nisourserver;
327     res_nmkupdate;
328     res_nupdate;
329     __res_no_hosts_fallback;
330     __res_opcodes;
331     __res_override_retry;
332     res_ownok;
333     __res_ownok;
334     res_pquery;
335     res_queriesmatch;
336     __res_queriesmatch;
337     res_rndinit;
338     res_randomid;
339     __res_randomid;
340     res_sendsigned;
341     res_sethostent;
342     __res_set_no_hosts_fallback;
343     __res_state;
344     __res_unset_no_hosts_fallback;
345     set_assertion_failure_callback;
346     sym_ntop;
347     __sym_ntop;
348     sym_ntos;
349     __sym_ntos;
350     sym_ston;
351     __sym_ston;
352     __tree_add;
353     __tree_init;
354     __tree_mung;
355     __tree_srch;
356     __tree_trav;
357     local:
358     *;
359 };

```

unchanged portion omitted

new/usr/src/lib/libresolv2/include/port_after.h

1

```
*****
14212 Sat Jan 31 23:14:47 2015
new/usr/src/lib/libresolv2/include/port_after.h
872 remove duplicate daemon() definitions
*****
1 /*
2  * Copyright (c) 1997, 2010, Oracle and/or its affiliates. All rights reserved.
3  */
4
5 /*
6  * Copyright (C) 2004-2008 Internet Systems Consortium, Inc. ("ISC")
7  * Copyright (C) 2001-2003 Internet Software Consortium.
8  *
9  * Permission to use, copy, modify, and/or distribute this software for any
10 * purpose with or without fee is hereby granted, provided that the above
11 * copyright notice and this permission notice appear in all copies.
12 *
13 * THE SOFTWARE IS PROVIDED "AS IS" AND ISC DISCLAIMS ALL WARRANTIES WITH
14 * REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
15 * AND FITNESS. IN NO EVENT SHALL ISC BE LIABLE FOR ANY SPECIAL, DIRECT,
16 * INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
17 * LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
18 * OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
19 * PERFORMANCE OF THIS SOFTWARE.
20 */
21
22 /* $Id: port_after.h.in,v 1.60 2008/02/28 05:34:17 marka Exp $ */
23
24 #ifndef port_after_h
25 #define port_after_h
26
27 #include <stdio.h>
28 #include <sys/types.h>
29 #include <sys/socket.h>
30 #include <sys/param.h>
31 #include <sys/time.h>
32 #if (!defined(BSD)) || (BSD < 199306)
33 #include <sys/bitypes.h>
34 #endif
35 #ifdef HAVE_INTTYPES_H
36 #include <inttypes.h>
37 #endif
38 #ifdef HAVE_SYS_SELECT_H
39 #include <sys/select.h>
40 #endif /* HAVE_SYS_SELECT_H */
41
42 #ifdef REENABLE_SEND
43 #undef send
44 #endif
45
46 #undef NEED_PSELECT
47 #undef HAVE_SA_LEN
48 #undef HAVE_MINIMUM_IFREQ
49 #define NEED_DAEMON 1
50 #undef NEED_STRSEP
51 #undef NEED_STRERROR
52 #ifdef NEED_STRERROR
53 const char *isc_strerror(int);
54 #define strerror isc_strerror
55 #endif
56 /* HAS_INET6_STRUCTS and HAVE_SIN6_SCOPE_ID are defined by port_ipv6.h
57 * #define HAS_INET6_STRUCTS 1
58 * #define HAVE_SIN6_SCOPE_ID 1
59 */
59 #include <port_ipv6.h>
```

new/usr/src/lib/libresolv2/include/port_after.h

2

```
61 #undef NEED_INET6ADDR_ANY
62 #undef HAS_IN_ADDR6
63 #define HAVE_SOCKETADDR_STORAGE 1
64 #undef NEED_GETTIMEOFDAY
65 #define HAVE_STRNDUP
66 #undef USE_FIONBIO_IOCTL
67 #undef INNETHGR_ARGS
68
69 #undef USE_IFNAMELINKID
70 #define PORT_NONBLOCK O_NONBLOCK
71
72 #ifndef _POSIX_PATH_MAX
73 #define _POSIX_PATH_MAX 255
74 #endif
75 #ifndef PATH_MAX
76 #define PATH_MAX _POSIX_PATH_MAX
77 #endif
78
79 /*
80 * We need to know the IPv6 address family number even on IPv4-only systems.
81 * Note that this is NOT a protocol constant, and that if the system has its
82 * own AF_INET6, different from ours below, all of BIND's libraries and
83 * executables will need to be recompiled after the system <sys/socket.h>
84 * has had this type added. The type number below is correct on most BSD-
85 * derived systems for which AF_INET6 is defined.
86 */
87 #ifndef AF_INET6
88 #define AF_INET6 24
89 #endif
90
91 #ifndef PF_INET6
92 #define PF_INET6 AF_INET6
93 #endif
94
95 #ifdef HAS_IN_ADDR6
96 /* Map to pre-RFC structure. */
97 #define in6_addr in_addr6
98 #endif
99
100 #ifndef HAS_INET6_STRUCTS
101 /* Replace with structure from later rev of O/S if known. */
102 struct in6_addr {
103     u_int8_t s6_addr[16];
104 };
105 #define unchanged_portion_omitted
106 #endif
107
108 #if !defined(HAS_INET6_STRUCTS) || defined(NEED_INET6ADDR_ANY)
109 #define in6addr_any isc_in6addr_any
110 extern const struct in6_addr in6addr_any;
111 #endif
112
113 /*
114 * IN6_ARE_ADDR_EQUAL, IN6_IS_ADDR_UNSPECIFIED, IN6_IS_ADDR_V4COMPAT and
115 * IN6_IS_ADDR_V4MAPPED are broken in glibc 2.1.
116 */
117 #ifdef __GLIBC__
118 #if __GLIBC__ < 2 || (__GLIBC__ == 2 && __GLIBC_MINOR__ < 2)
119 #undef IN6_ARE_ADDR_EQUAL
120 #undef IN6_IS_ADDR_UNSPECIFIED
121 #undef IN6_IS_ADDR_V4COMPAT
122 #undef IN6_IS_ADDR_V4MAPPED
123 #endif
124 #endif
```



```

210 #ifndef IN6_ARE_ADDR_EQUAL
211 #define IN6_ARE_ADDR_EQUAL(a,b) \
212     (memcmp(&(a)->s6_addr[0], &(b)->s6_addr[0], sizeof(struct in6_addr)) == 0)
213 #endif

215 #ifndef IN6_IS_ADDR_UNSPECIFIED
216 #define IN6_IS_ADDR_UNSPECIFIED(a) \
217     IN6_ARE_ADDR_EQUAL(a, &in6addr_any)
218 #endif

220 #ifndef IN6_IS_ADDR_LOOPBACK
221 extern const struct in6_addr isc_in6addr_loopback;
222 #define IN6_IS_ADDR_LOOPBACK(a) \
223     IN6_ARE_ADDR_EQUAL(a, &isc_in6addr_loopback)
224 #endif

226 #ifndef IN6_IS_ADDR_V4MAPPED
227 #define IN6_IS_ADDR_V4MAPPED(a) \
228     ((a)->s6_addr[0] == 0x00 && (a)->s6_addr[1] == 0x00 && \
229     (a)->s6_addr[2] == 0x00 && (a)->s6_addr[3] == 0x00 && \
230     (a)->s6_addr[4] == 0x00 && (a)->s6_addr[5] == 0x00 && \
231     (a)->s6_addr[6] == 0x00 && (a)->s6_addr[9] == 0x00 && \
232     (a)->s6_addr[8] == 0x00 && (a)->s6_addr[9] == 0x00 && \
233     (a)->s6_addr[10] == 0xff && (a)->s6_addr[11] == 0xff)
234 #endif

236 #ifndef IN6_IS_ADDR_SITELocal
237 #define IN6_IS_ADDR_SITELocal(a) \
238     (((a)->s6_addr[0] == 0xfe) && (((a)->s6_addr[1] & 0xc0) == 0xc0))
239 #endif

241 #ifndef IN6_IS_ADDR_LINKLOCAL
242 #define IN6_IS_ADDR_LINKLOCAL(a) \
243     (((a)->s6_addr[0] == 0xfe) && (((a)->s6_addr[1] & 0xc0) == 0x80))
244 #endif

246 #ifndef IN6_IS_ADDR_MULTICAST
247 #define IN6_IS_ADDR_MULTICAST(a) \
248     ((a)->s6_addr[0] == 0xff)

250 #ifndef __IPV6_ADDR_MC_SCOPE
251 #define __IPV6_ADDR_MC_SCOPE(a) \
252     ((a)->s6_addr[1] & 0x0f)
253 #endif

254 #ifndef __IPV6_ADDR_SCOPE_SITELocal
255 #define __IPV6_ADDR_SCOPE_SITELocal 0x05
256 #endif
257 #ifndef __IPV6_ADDR_SCOPE_ORGLocal
258 #define __IPV6_ADDR_SCOPE_ORGLocal 0x08
259 #endif

261 #ifndef IN6_IS_ADDR_MC_SITELocal
262 #define IN6_IS_ADDR_MC_SITELocal(a) \
263     (IN6_IS_ADDR_MULTICAST(a) && \
264     (__IPV6_ADDR_MC_SCOPE(a) == __IPV6_ADDR_SCOPE_SITELocal))
265 #endif

267 #ifndef IN6_IS_ADDR_MC_ORGLocal
268 #define IN6_IS_ADDR_MC_ORGLocal(a) \
269     (IN6_IS_ADDR_MULTICAST(a) && \
270     (__IPV6_ADDR_MC_SCOPE(a) == __IPV6_ADDR_SCOPE_ORGLocal))
271 #endif

273 #ifndef INADDR_NONE
274 #define INADDR_NONE 0xffffffff
275 #endif

```

```

277 #ifndef MAXHOSTNAMELEN
278 #define MAXHOSTNAMELEN 256
279 #endif

281 #ifndef INET6_ADDRSTRLEN
282 /* sizeof("aaaa:bbbb:cccc:dddd:eeee:ffff:123.123.123.123") */
283 #define INET6_ADDRSTRLEN 46
284 #endif

286 #ifndef MIN
287 #define MIN(x,y) (((x) <= (y)) ? (x) : (y))
288 #endif

290 #ifndef MAX
291 #define MAX(x,y) (((x) >= (y)) ? (x) : (y))
292 #endif

295 #ifndef NEED_DAEMON
296 int daemon(int nochdir, int noclose);
297 #endif

299 #ifndef NEED_STRSEP
299 char * strsep(char **stringp, const char *delimiter);
296 #endif

298 #ifndef ALIGN
299 #define ALIGN(p) (((uintptr_t)(p) + (sizeof(long) - 1)) & ~(sizeof(long) - 1))
300 #endif

302 #ifndef NEED_SETGROUPENT
303 int setgroupent(int stayopen);
304 #endif

306 #ifndef NEED_GETGROUPLIST
307 int getgrouplist(GETGROUPLIST_ARGS);
308 #endif

310 #ifndef POSIX_GETGRNAM_R
311 int
312 __posix_getgrnam_r(const char *, struct group *, char *, int, struct group **);
313 #endif

315 #ifndef NEED_GETGRNAM_R
316 int
317 getgrnam_r(const char *, struct group *, char *, size_t, struct group **);
318 #endif

320 #ifndef POSIX_GETGRGID_R
321 int
322 __posix_getgrgid_r(gid_t, struct group *, char *, int, struct group **);
323 #endif

325 #ifndef NEED_GETGRGID_R
326 int
327 getgrgid_r(gid_t, struct group *, char *, size_t, struct group **);
328 #endif

330 #ifndef NEED_GETGRENT_R
331 GROUP_R_RETURN getgrent_r(struct group *gp, GROUP_R_ARGS);
332 #endif

334 #ifndef NEED_SETGRENT_R
335 GROUP_R_SET_RETURN setgrent_r(GROUP_R_ENT_ARGS);
336 #endif

```

```

338 #ifdef NEED_ENDGRENT_R
339 GROUP_R_END_RETURN endgrent_r(GROUP_R_ENT_ARGS);
340 #endif

342 #if defined(NEED_INNETGR_R) && defined(NGR_R_RETURN)
343 NGR_R_RETURN
344 innetgr_r(const char *, const char *, const char *, const char *);
345 #endif

347 #ifdef NEED_SETNETGRENT_R
348 #ifdef NGR_R_SET_ARGS
349 NGR_R_SET_RETURN setnetgrent_r(NGR_R_SET_CONST char *netgroup, NGR_R_SET_ARGS);
350 #else
351 NGR_R_SET_RETURN setnetgrent_r(NGR_R_SET_CONST char *netgroup);
352 #endif
353 #endif

355 #ifdef NEED_ENDNETGRENT_R
356 #ifdef NGR_R_END_ARGS
357 NGR_R_END_RETURN endnetgrent_r(NGR_R_END_ARGS);
358 #else
359 NGR_R_END_RETURN endnetgrent_r(void);
360 #endif
361 #endif

363 #ifdef POSIX_GETPWNAM_R
364 int
365 __posix_getpwnam_r(const char *login, struct passwd *pwptr,
366                  char *buf, size_t buflen, struct passwd **result);
367 #endif

369 #ifdef NEED_GETPWNAM_R
370 int
371 getpwnam_r(const char *login, struct passwd *pwptr,
372            char *buf, size_t buflen, struct passwd **result);
373 #endif

375 #ifdef POSIX_GETPWUID_R
376 int
377 __posix_getpwuid_r(uid_t uid, struct passwd *pwptr,
378                  char *buf, int buflen, struct passwd **result);
379 #endif

381 #ifdef NEED_GETPWUID_R
382 int
383 getpwuid_r(uid_t uid, struct passwd *pwptr,
384            char *buf, size_t buflen, struct passwd **result);
385 #endif

387 #ifdef NEED_SETPWENT_R
388 #ifdef PASS_R_ENT_ARGS
389 PASS_R_SET_RETURN setpwent_r(PASS_R_ENT_ARGS);
390 #else
391 PASS_R_SET_RETURN setpwent_r(void);
392 #endif

394 #endif

396 #ifdef NEED_SETPASSENT_R
397 #ifdef PASS_R_ENT_ARGS
398 PASS_R_SET_RETURN setpassent_r(int stayopen, PASS_R_ENT_ARGS);
399 #else
400 PASS_R_SET_RETURN setpassent_r(int stayopen);
401 #endif
402 #endif

```

```

404 #ifdef NEED_GETPWENT_R
405 PASS_R_RETURN getpwent_r(struct passwd *pwptr, PASS_R_ARGS);
406 #endif

408 #ifdef NEED_ENDPWENT_R
409 void endpwent_r(void);
410 #endif

412 #ifdef NEED_SETPASSENT
413 int setpassent(int stayopen);
414 #endif

416 #define gettimeofday isc__gettimeofday
417 #ifdef NEED_GETTIMEOFDAY
418 int isc__gettimeofday(struct timeval *tvp, struct _TIMEZONE *tzp);
419 #else
420 int isc__gettimeofday(struct timeval *tp, struct timezone *tzp);
421 #endif

423 int getnetgrent(NGR_R_CONST char **machinep, NGR_R_CONST char **userp,
424               NGR_R_CONST char **domainp);

426 #ifdef NGR_R_ARGS
427 int getnetgrent_r(NGR_R_CONST char **machinep, NGR_R_CONST char **userp,
428                 NGR_R_CONST char **domainp, NGR_R_ARGS);
429 #endif

431 /* setnetgrent and endnetgrent are defined in sunw_port_after.h
432 #ifdef SETNETGRENT_ARGS
433 void setnetgrent(SETNETGRENT_ARGS);
434 #else
435 void setnetgrent(const char *netgroup);
436 #endif

438 void endnetgrent(void);
439 */

441 #ifdef INNETGR_ARGS
442 int innetgr(INNETGR_ARGS);
443 #else
444 int innetgr(const char *netgroup, const char *machine,
445            const char *user, const char *domain);
446 #endif

448 #ifdef NGR_R_SET_ARGS
449 NGR_R_SET_RETURN
450 setnetgrent_r(NGR_R_SET_CONST char *netgroup, NGR_R_SET_ARGS);
451 #else
452 NGR_R_SET_RETURN
453 setnetgrent_r(NGR_R_SET_CONST char *netgroup);
454 #endif

456 #ifdef NEED_STRTOUL
457 unsigned long strtoul(const char *, char **, int);
458 #endif

460 #ifdef NEED_SUN4PROTOS
461 #include <stdarg.h>
462 #ifndef __SIZE_TYPE__
463 #define __SIZE_TYPE__ int
464 #endif
465 struct sockaddr;
466 struct iovec;
467 struct timeval;
468 struct timezone;
469 int fprintf(FILE *, const char *, ...);

```

```
470 int getsockname(int, struct sockaddr *, int *);
471 int getpeername(int, struct sockaddr *, int *);
472 int socket(int, int, int);
473 int connect(int, const struct sockaddr *, int);
474 int writev(int, struct iovec *, int);
475 int readv(int, struct iovec *, int);
476 int send(int, const char *, int, int);
477 void bzero(char *, int);
478 int recvfrom(int, char *, int, int, struct sockaddr *, int *);
479 int syslog(int, const char *, ... );
480 int printf(const char *, ...);
481 __SIZE_TYPE__ fread(void *, __SIZE_TYPE__, __SIZE_TYPE__, FILE *);
482 __SIZE_TYPE__ fwrite(const void *, __SIZE_TYPE__, __SIZE_TYPE__, FILE *);
483 int fclose(FILE *);
484 int ungetc(int, FILE *);
485 int scanf(const char *, ...);
486 int sscanf(const char *, const char *, ... );
487 int tolower(int);
488 int toupper(int);
489 int strcasecmp(const char *, const char *);
490 int strncasecmp(const char *, const char *, int);
491 int select(int, fd_set *, fd_set *, fd_set *, struct timeval *);
492 #ifdef gettimeofday
493 #undef gettimeofday
494 int gettimeofday(struct timeval *, struct timezone *);
495 #define gettimeofday isc__gettimeofday
496 #else
497 int gettimeofday(struct timeval *, struct timezone *);
498 #endif
499 long strtol(const char *, char **, int);
500 int fseek(FILE *, long, int);
501 int setsockopt(int, int, int, const char *, int);
502 int bind(int, const struct sockaddr *, int);
503 void bcopy(char *, char *, int);
504 int fputc(char, FILE *);
505 int listen(int, int);
506 int accept(int, struct sockaddr *, int *);
507 int getsockopt(int, int, int, char *, int *);
508 int vfprintf(FILE *, const char *, va_list);
509 int fflush(FILE *);
510 int fgetc(FILE *);
511 int fputs(const char *, FILE *);
512 int fchown(int, int, int);
513 void setbuf(FILE *, char *);
514 int gethostname(char *, int);
515 int rename(const char *, const char *);
516 time_t time(time_t *);
517 int fscanf(FILE *, const char *, ...);
518 int sscanf(const char *, const char *, ...);
519 int ioctl(int, int, caddr_t);
520 void perror(const char *);

522 #if !defined(__USE_FIXED_PROTOTYPES__) && !defined(__cplusplus) && !defined(__ST
523 /*
524  * 'gcc -ansi' changes the prototype for vsprintf().
525  * Use this prototype when 'gcc -ansi' is not in effect.
526  */
527 char *vsprintf(char *, const char *, va_list);
528 #endif
529 #endif

531 /* Solaris-specific changes */
532 #include "sunw_port_after.h"

534 #endif /* port_after_h */
```