

new/usr/src/Makefile.master

1

```
*****
38165 Thu Apr 25 02:20:38 2019
new/usr/src/Makefile.master
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 #
23 # Copyright (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright (c) 2012 by Delphix. All rights reserved.
25 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
26 # Copyright 2015, OmniTI Computer Consulting, Inc. All rights reserved.
27 # Copyright 2015 Gary Mills
28 # Copyright 2015 Igor Kozhukhov <ikozhukhov@gmail.com>
29 # Copyright 2016 Toomas Soome <tsoome@me.com>
30 # Copyright 2018 OmniOS Community Edition (OmniOSce) Association.
31 # Copyright (c) 2019, Joyent, Inc.
32 #
33 #
34 #
35 # Makefile.master, global definitions for system source
36 #
37 ROOT=          /proto
38 #
39 #
40 # Adjunct root, containing an additional proto area to be used for headers
41 # and libraries.
42 #
43 ADJUNCT_PROTO=
44 #
45 #
46 # Adjunct for building things that run on the build machine.
47 #
48 NATIVE_ADJUNCT= /usr
49 #
50 #
51 # RELEASE_BUILD should be cleared for final release builds.
52 # NOT_RELEASE_BUILD is exactly what the name implies.
53 #
```

new/usr/src/Makefile.master

2

```
54 # __GNUC toggles the building of ON components using gcc and related tools.
55 # Normally set to '#', set it to '' to do gcc build.
56 #
57 # The declaration POUND_SIGN is always '#'. This is needed to get around the
58 # make feature that '#' is always a comment delimiter, even when escaped or
59 # quoted. We use this macro expansion method to get POUND_SIGN rather than
60 # always breaking out a shell because the general case can cause a noticeable
61 # slowdown in build times when so many Makefiles include Makefile.master.
62 #
63 # While the majority of users are expected to override the setting below
64 # with an env file (via nightly or bldenv), if you aren't building that way
65 # (ie, you're using "ws" or some other bootstrapping method) then you need
66 # this definition in order to avoid the subshell invocation mentioned above.
67 #
68 #
69 PRE_POUND=      pre\#
70 POUND_SIGN=     $(PRE_POUND:pre\#=%)
71 #
72 NOT_RELEASE_BUILD=
73 RELEASE_BUILD=  $(POUND_SIGN)
74 $(RELEASE_BUILD)NOT_RELEASE_BUILD= $(POUND_SIGN)
75 PATCH_BUILD=    $(POUND_SIGN)
76 #
77 # SPARC_BLD is '#' for an Intel build.
78 # INTEL_BLD is '#' for a Sparc build.
79 SPARC_BLD_1=    $(MACH:i386=$(POUND_SIGN))
80 SPARC_BLD=      $(SPARC_BLD_1:sparc=)
81 INTEL_BLD_1=    $(MACH:sparc=$(POUND_SIGN))
82 INTEL_BLD=      $(INTEL_BLD_1:i386=)
83 #
84 # The variables below control the compilers used during the build.
85 # There are a number of permutations.
86 #
87 # __GNUC and __SUNC control (and indicate) the primary compiler. Whichever
88 # one is not POUND_SIGN is the primary, with the other as the shadow. They
89 # may also be used to control entirely compiler-specific Makefile assignments.
90 # __GNUC and GCC are the default.
91 #
92 # __GNUC64 indicates that the 64bit build should use the GNU C compiler.
93 # There is no Sun C analogue.
94 #
95 # The following version-specific options are operative regardless of which
96 # compiler is primary, and control the versions of the given compilers to be
97 # used. They also allow compiler-version specific Makefile fragments.
98 #
99 #
100 __SUNC=          $(POUND_SIGN)
101 $(__SUNC)__GNUC= $(POUND_SIGN)
102 __GNUC64=        $(__GNUC)
103 #
104 # Allow build-time "configuration" to enable or disable some things.
105 # The default is POUND_SIGN, meaning "not enabled". If the environment
106 # passes in an override like ENABLE_SMB_PRINTING= (empty) that will
107 # uncomment things in the lower Makefiles to enable the feature.
108 ENABLE_SMB_PRINTING= $(POUND_SIGN)
109 #
110 # CLOSED is the root of the tree that contains source which isn't released
111 # as open source
112 CLOSED=          $(SRC)/../closed
113 #
114 # BUILD_TOOLS is the root of all tools including compilers.
115 # ONBLD_TOOLS is the root of all the tools that are part of SUNWonbld.
116 #
117 BUILD_TOOLS=     /ws/onnv-tools
118 ONBLD_TOOLS=     $(BUILD_TOOLS)/onbld
```

```

120 # define runtime JAVA_HOME, primarily for cmd/pools/poold
121 JAVA_HOME= /usr/java
122 # define buildtime JAVA_ROOT
123 JAVA_ROOT= /usr/java
124 # Build uses java7 by default. Pass one the variables below set to empty
125 # string in the environment to override.
126 BLD_JAVA_6= $(POUND_SIGN)
127 BLD_JAVA_8= $(POUND_SIGN)

129 GNUC_ROOT= /opt/gcc/4.4.4
130 GCCLIBDIR= $(GNUMC_ROOT)/lib
131 GCCLIBDIR64= $(GNUMC_ROOT)/lib/$(MACH64)

133 DOCBOOK_XSL_ROOT= /usr/share/sgml/docbook/xsl-stylesheets

135 RPCGEN= /usr/bin/rpcgen
136 STABS= $(ONBLD_TOOLS)/bin/$(MACH)/stabs
137 ELFEXTRACT= $(ONBLD_TOOLS)/bin/$(MACH)/elfextract
138 MBH_PATCH= $(ONBLD_TOOLS)/bin/$(MACH)/mbh_patch
139 BTXLD= $(ONBLD_TOOLS)/bin/$(MACH)/btxld
140 VTFONTCVT= $(ONBLD_TOOLS)/bin/$(MACH)/vtfontcvt
141 # echo(1) and true(1) are specified without absolute paths, so that the shell
142 # spawned by make(1) may use the built-in versions. This is minimally
143 # problematic, as the shell spawned by make(1) is known and under control, the
144 # only risk being if the shell falls back to $PATH.
145 #
146 # We specifically want an echo(1) that does interpolation of escape sequences,
147 # which ksh93, /bin/sh, and bash will all provide.
148 ECHO= echo
149 TRUE= true
150 INS= $(ONBLD_TOOLS)/bin/$(MACH)/install
151 SYMLINK= /usr/bin/ln -s
152 LN= /usr/bin/ln
153 MKDIR= /usr/bin/mkdir
154 CHMOD= /usr/bin/chmod
155 MV= /usr/bin/mv -f
156 RM= /usr/bin/rm -f
157 CUT= /usr/bin/cut
158 NM= /usr/ccs/bin/nm
159 DIFF= /usr/bin/diff
160 GREP= /usr/bin/grep
161 EGREP= /usr/bin/egrep
162 ELFWRAP= /usr/bin/elfwrap
163 KSH93= /usr/bin/ksh93
164 SED= /usr/bin/sed
165 AWK= /usr/bin/nawk
166 CP= /usr/bin/cp -f
167 MCS= /usr/ccs/bin/mcs
168 CAT= /usr/bin/cat
169 ELFDUMP= /usr/ccs/bin/elfdump
170 M4= /usr/bin/m4
171 GM4= /usr/bin/gm4
172 STRIP= /usr/ccs/bin/strip
173 LEX= /usr/ccs/bin/lex
174 FLEX= /usr/bin/flex
175 YACC= /usr/ccs/bin/yacc
176 BISON= /usr/bin/bison
177 CPP= /usr/lib/cpp
178 ANSI_CPP= $(GNUMC_ROOT)/bin/cpp
179 JAVAC= $(JAVA_ROOT)/bin/javac
180 JAVAH= $(JAVA_ROOT)/bin/javah
181 JAVADOC= $(JAVA_ROOT)/bin/javadoc
182 RMIC= $(JAVA_ROOT)/bin/rmic
183 JAR= $(JAVA_ROOT)/bin/jar
184 CTFCONVERT= $(ONBLD_TOOLS)/bin/$(MACH)/ctfconvert
185 CTFDIFF= $(ONBLD_TOOLS)/bin/$(MACH)/ctfdiff

```

```

186 CTFMERGE= $(ONBLD_TOOLS)/bin/$(MACH)/ctfmerge
187 CTFSTABS= $(ONBLD_TOOLS)/bin/$(MACH)/ctfstabs
188 CTFSTRIP= $(ONBLD_TOOLS)/bin/$(MACH)/ctfststrip
189 NDRGEN= $(ONBLD_TOOLS)/bin/$(MACH)/ndrgen
190 GENOFFSETS= $(ONBLD_TOOLS)/bin/genoffsets
191 XREF= $(ONBLD_TOOLS)/bin/xref
192 FIND= /usr/bin/find
193 PERL= /usr/bin/perl
194 PERL_VERSION= 5.10.0
195 PERL_PKGVERS= -510
196 PERL_ARCH = i86pc-solaris-64int
197 $(SPARC_BLD)PERL_ARCH = sun4-solaris-64int
198 PYTHON_VERSION= 2.7
199 PYTHON_PKGVERS= -27
200 PYTHON_SUFFIX=
201 PYTHON= /usr/bin/python$(PYTHON_VERSION)
202 PYTHON3_VERSION= 3.5
203 PYTHON3_PKGVERS= -35
204 PYTHON3_SUFFIX= m
205 PYTHON3= /usr/bin/python$(PYTHON3_VERSION)
206 $(BUILDPY3TOOLS)TOOLS_PYTHON= $(PYTHON3)
207 $(BUILDPY2TOOLS)TOOLS_PYTHON= $(PYTHON)
208 SORT= /usr/bin/sort
209 TR= /usr/bin/tr
210 TOUCH= /usr/bin/touch
211 WC= /usr/bin/wc
212 XARGS= /usr/bin/xargs
213 ELFEDIT= /usr/bin/elfedit
214 DTRACE= /usr/sbin/dtrace -xnolib
215 UNIQ= /usr/bin/uniq
216 TAR= /usr/bin/tar
217 ASTBINDIR= /usr/ast/bin
218 MSGCC= $(ASTBINDIR)/msgcc
219 MSGFMT= /usr/bin/msgfmt -s
220 LCDEF= $(ONBLD_TOOLS)/bin/$(MACH)/localedef
221 TIC= $(ONBLD_TOOLS)/bin/$(MACH)/tic
222 ZIC= $(ONBLD_TOOLS)/bin/$(MACH)/zic
223 OPENSSEL= /usr/bin/openssl
224 CPCGEN= $(ONBLD_TOOLS)/bin/$(MACH)/cpcgen

226 FILEMODE= 644
227 DIRMODE= 755

229 # Declare that nothing should be built in parallel.
230 # Individual Makefiles can use the .PARALLEL target to declare otherwise.
231 .NO_PARALLEL:

233 # For stylistic checks
234 #
235 # Note that the X and C checks are not used at this time and may need
236 # modification when they are actually used.
237 #
238 CSTYLE= $(ONBLD_TOOLS)/bin/cstyle
239 CSTYLE_TAIL=
240 HDRCHK= $(ONBLD_TOOLS)/bin/hdrchk
241 HDRCHK_TAIL=
242 JSTYLE= $(ONBLD_TOOLS)/bin/jstyle

244 DOT_H_CHECK= \
245 @$(ECHO) "checking $<"; $(CSTYLE) $< $(CSTYLE_TAIL); \
246 $(HDRCHK) $< $(HDRCHK_TAIL)

248 DOT_X_CHECK= \
249 @$(ECHO) "checking $<"; $(RPCGEN) -C -h $< | $(CSTYLE) $(CSTYLE_TAIL); \
250 $(RPCGEN) -C -h $< | $(HDRCHK) $< $(HDRCHK_TAIL)

```

```

252 DOT_C_CHECK= \
253     @$ (ECHO) "checking $<"; $(CSTYLE) $< $(CSTYLE_TAIL)

255 MANIFEST_CHECK= \
256     @$ (ECHO) "checking $<"; \
257     SVCCFG_DTD=$(SRC)/cmd/svc/dtd/service_bundle.dtd.1 \
258     SVCCFG_REPOSITORY=$(SRC)/cmd/svc/seed/global.db \
259     SVCCFG_CONFIGD_PATH=$(SRC)/cmd/svc/configd/svc.configd-native \
260     $(SRC)/cmd/svc/svccfg/svccfg-native validate $<

262 INS.file=      $(RM) $@; $(INS) -s -m $(FILEMODE) -f $(@D) $<
263 INS.dir=      $(INS) -s -d -m $(DIRMODE) $@
264 # installs and renames at once
265 #
266 INS.rename=   $(INS.file); $(MV) $(@D)/$(<F) $@

268 # install a link
269 INSLINKTARGET= $<
270 INS.link=     $(RM) $@; $(LN) $(INSLINKTARGET) $@
271 INS.symlink=  $(RM) $@; $(SYMLINK) $(INSLINKTARGET) $@

273 # The path to python that will be used for the shebang line when installing
274 # python scripts to the proto area. This is overridden by makefiles to
275 # select to the correct version.
276 PYSHEBANG=   $(PYTHON)

278 #
279 # Python bakes the mtime of the .py file into the compiled .pyc and
280 # rebuilds if the baked-in mtime != the mtime of the source file
281 # (rather than only if it's less than), thus when installing python
282 # files we must make certain to not adjust the mtime of the source
283 # (.py) file.
284 #
285 INS.pyfile=   $(RM) $@; $(SED) \
286     -e "1s:^\#!@PYTHON@:\#!$(PYSHEBANG):" \
287     -e "1s:^\#!@TOOLS_PYTHON@:\#!$(TOOLS_PYTHON):" \
288     < $< > $@; $(CHMOD) $(FILEMODE) $@; $(TOUCH) -r $< $@

290 # MACH must be set in the shell environment per uname -p on the build host
291 # More specific architecture variables should be set in lower makefiles.
292 #
293 # MACH64 is derived from MACH, and BUILD64 is set to '#' for
294 # architectures on which we do not build 64-bit versions.
295 # (There are no such architectures at the moment.)
296 #
297 # Set BUILD64=# in the environment to disable 64-bit amd64
298 # builds on i386 machines.

300 MACH64_1=     $(MACH:sparc=sparcv9)
301 MACH64=      $(MACH64_1:i386=amd64)

303 MACH32_1=    $(MACH:sparc=sparcv7)
304 MACH32=      $(MACH32_1:i386=i86)

306 sparc_BUILD64=
307 i386_BUILD64=
308 BUILD64=     $($ (MACH)_BUILD64)

310 #
311 # C compiler mode. Future compilers may change the default on us,
312 # so force extended ANSI mode globally. Lower level makefiles can
313 # override this by setting CCMODE.
314 #
315 CCMODE=      -Xa
316 CCMODE64=   -Xa

```

```

318 #
319 # C compiler verbose mode. This is so we can enable it globally,
320 # but turn it off in the lower level makefiles of things we cannot
321 # (or aren't going to) fix.
322 #
323 CVERBOSE=    -v

325 # set this to the secret flag "-Wc,-Qiselect-v9abiwarn=1" to get warnings
326 # from the compiler about places the -xarch=v9 may differ from -xarch=v9c.
327 V9ABIWARN=

329 # set this to the secret flag "-Wc,-Qiselect-regsym=0" to disable register
330 # symbols (used to detect conflicts between objects that use global registers)
331 # we disable this now for safety, and because genunix doesn't link with
332 # this feature (the v9 default) enabled.
333 #
334 # REGSYM is separate since the C++ driver syntax is different.
335 CCREGSYM=    -Wc,-Qiselect-regsym=0
336 CCCREGSYM=   -Qoption cg -Qiselect-regsym=0

338 # Prevent the removal of static symbols by the SPARC code generator (cg).
339 # The x86 code generator (ube) does not remove such symbols and as such
340 # using this workaround is not applicable for x86.
341 #
342 CCSTATICSYM= -Wc,-Qassembler-ounrefsym=0
343 #
344 # generate 32-bit addresses in the v9 kernel. Saves memory.
345 CCABS32=     -Wc,-xcode=abs32
346 #
347 # generate v9 code which tolerates callers using the v7 ABI, for the sake of
348 # system calls.
349 CC32BITCALLERS= -_gcc=-massume-32bit-callers

351 # GCC, especially, is increasingly beginning to auto-inline functions and
352 # sadly does so separately not under the general -fno-inline-functions
353 # Additionally, we wish to prevent optimisations which cause GCC to clone
354 # functions -- in particular, these may cause unhelpful symbols to be
355 # emitted instead of function names
356 CCNOAUTOINLINE= \
357     -_gcc=-fno-inline-small-functions \
358     -_gcc=-fno-inline-functions-called-once \
359     -_gcc=-fno-ipa-cp \
360     -_gcc7=-fno-ipa-icf \
361     -_gcc8=-fno-ipa-icf \
362     -_gcc7=-fno-clone-functions \
363     -_gcc8=-fno-clone-functions

365 # GCC may put functions in different named sub-sections of .text based on
366 # their presumed calling frequency. At least in the kernel, where we actually
367 # deliver relocatable objects, we don't want this to happen.
368 #
369 # Since at present we don't benefit from this even in userland, we disable it gl
370 # but the application of this may move into usr/src/uts/ in future.
371 CCNOREORDER= \
372     -_gcc7=-fno-reorder-functions \
373     -_gcc8=-fno-reorder-functions

375 #
376 # gcc has a rather aggressive optimization on by default that infers loop
377 # bounds based on undefined behavior (!!). This can lead to some VERY
378 # surprising optimizations -- ones that may be technically correct in the
379 # strictest sense but also result in incorrect program behavior. We turn
380 # this optimization off, with extreme prejudice.
381 #
382 CCNOAGGRESSIVELOOPS= \
383     -_gcc7=-fno-aggressive-loop-optimizations \

```

```

384      _gcc8=-fno-aggressive-loop-optimizations

386 # One optimization the compiler might perform is to turn this:
387 #   #pragma weak foo
388 #   extern int foo;
389 #   if (&foo)
390 #       foo = 5;
391 # into
392 #   foo = 5;
393 # Since we do some of this (foo might be referenced in common kernel code
394 # but provided only for some cpu modules or platforms), we disable this
395 # optimization.
396 #
397 sparc_CCUNBOUND = -Wd,-xsafe=unboundsym
398 i386_CCUNBOUND  =
399 CCUNBOUND       = $($ (MACH)_CCUNBOUND)

401 #
402 # compiler '-xarch' flag. This is here to centralize it and make it
403 # overridable for testing.
404 sparc_XARCH=      -m32
405 sparcv9_XARCH=   -m64
406 i386_XARCH=      -m32
407 amd64_XARCH=     -m64 -Ui386 -U__i386

409 # assembler '-xarch' flag. Different from compiler '-xarch' flag.
410 sparc_AS_XARCH=   -xarch=v8plus
411 sparcv9_AS_XARCH= -xarch=v9
412 i386_AS_XARCH=
413 amd64_AS_XARCH=   -xarch=amd64 -P -Ui386 -U__i386

415 #
416 # These flags define what we need to be 'standalone' i.e. -not- part
417 # of the rather more cosy userland environment. This basically means
418 # the kernel.
419 #
420 # XX64 future versions of gcc will make -mmodel=kernel imply -mno-red-zone
421 #
422 sparc_STAND_FLAGS=  _gcc=-ffreestanding
423 sparcv9_STAND_FLAGS= _gcc=-ffreestanding
424 # Disabling MMX also disables 3DNow, disabling SSE also disables all later
425 # additions to SSE (SSE2, AVX ,etc.)
426 NO_SIMD=           _gcc=-mno-mmx _gcc=-mno-sse
427 i386_STAND_FLAGS=  _gcc=-ffreestanding $($ (NO_SIMD))
428 amd64_STAND_FLAGS= -xmodel=kernel $($ (NO_SIMD))

430 SAVEARGS=          -Wu,-save_args
431 amd64_STAND_FLAGS += $($ (SAVEARGS))

433 STAND_FLAGS_32 = $($ (MACH)_STAND_FLAGS)
434 STAND_FLAGS_64 = $($ (MACH64)_STAND_FLAGS)

436 #
437 # disable the incremental linker
438 ILDOFF=            -xildoff
439 #
440 XFFLAG=           -xF=%all
441 XESS=             -xs
442 XSTRCONST=       -xstrconst

444 #
445 # turn warnings into errors (C)
446 CERRWARN = -errtags=yes -errwarn=all
447 CERRWARN += -erroff=E_EMPTY_TRANSLATION_UNIT
448 CERRWARN += -erroff=E_STATEMENT_NOT_REACHED

```

```

450 CERRWARN += _gcc=-Wno-missing-braces
451 CERRWARN += _gcc=-Wno-sign-compare
452 CERRWARN += _gcc=-Wno-unknown-pragmas
453 CERRWARN += _gcc=-Wno-unused-parameter
454 CERRWARN += _gcc=-Wno-missing-field-initializers

456 # Unfortunately, this option can misfire very easily and unfixably.
457 CERRWARN += _gcc=-Wno-array-bounds

460 CERRWARN += _smatch=-p=illumos_user
461 include $(SRC)/Makefile.smatch

463 #
464 # turn warnings into errors (C++)
465 CCERRWARN=        -xwe

467 # C standard. Keep Studio flags until we get rid of lint.
468 CSTD_GNU89=       -xc99=%none
469 CSTD_GNU99=       -xc99=%all
470 CSTD=             $($ (CSTD_GNU89))
471 C99LMODE=         $($ (CSTD:-xc99%=-Xc99%))

473 # In most places, assignments to these macros should be appended with +=
474 # (CPPFLAGS.first allows values to be prepended to CPPFLAGS).
475 sparc_CFLAGS=     $($ (sparc_XARCH) $($ (CCSTATICSYM))
476 sparcv9_CFLAGS=   $($ (sparcv9_XARCH) -dalign $($ (CCVERBOSE)) $($ (V9ABIWARN)) $($ (CCREGSYM)) \
477                 $($ (CCSTATICSYM))
478 i386_CFLAGS=      $($ (i386_XARCH))
479 amd64_CFLAGS=     $($ (amd64_XARCH))

481 sparc_ASFLAGS=    $($ (sparc_AS_XARCH))
482 sparcv9_ASFLAGS=  $($ (sparcv9_AS_XARCH))
483 i386_ASFLAGS=     $($ (i386_AS_XARCH))
484 amd64_ASFLAGS=    $($ (amd64_AS_XARCH))

486 #
487 sparc_COPTFLAG=    -xO3
488 sparcv9_COPTFLAG= -xO3
489 i386_COPTFLAG=    -O
490 amd64_COPTFLAG=   -xO3

492 COPTFLAG=         $($ (MACH)_COPTFLAG)
493 COPTFLAG64=       $($ (MACH64)_COPTFLAG)

495 # When -g is used, the compiler globalizes static objects
496 # (gives them a unique prefix). Disable that.
497 CNOGLOBAL=        -W0,-noglobal

499 # Direct the Sun Studio compiler to use a static globalization prefix based on t
500 # name of the module rather than something unique. Otherwise, objects
501 # will not build deterministically, as subsequent compilations of identical
502 # source will yeild objects that always look different.
503 #
504 # In the same spirit, this will also remove the date from the N_OPT stab.
505 CGLOBALSTATIC=    -W0,-xglobalstatic

507 # Sometimes we want all symbols and types in debugging information even
508 # if they aren't used.
509 CALLSYMS=         -W0,-xdbggen=no%usedonly

511 #
512 # We force the compilers to generate the debugging information best understood
513 # by the CTF tools. With Sun Studio this is stabs due to bugs in the Studio
514 # compilers. With GCC this is DWARF v2.
515 #

```

```

516 DEBUGFORMAT=      _cc=-xdebugformat=stabs _gcc=-gdwarf-2
518 #
519 # Ask the compiler to include debugging information
520 #
521 CCGDEBUG= -g $(DEBUGFORMAT)
523 #
524 # Flags used to build in debug mode for ctf generation.
525 #
526 CTF_FLAGS_sparc = $(CCGDEBUG) -Wc,-Qiselect-T1 $(CSTD) $(CNOGLOBAL)
527 CTF_FLAGS_i386  = $(CCGDEBUG) $(CSTD) $(CNOGLOBAL)
529 CTF_FLAGS_sparcv9      = $(CTF_FLAGS_sparc)
530 CTF_FLAGS_amd64       = $(CTF_FLAGS_i386)
532 # Sun Studio produces broken userland code when saving arguments.
533 $(__GNUCC)CTF_FLAGS_amd64 += $(SAVEARGS)
535 CTF_FLAGS_32          = $(CTF_FLAGS_$(MACH))
536 CTF_FLAGS_64         = $(CTF_FLAGS_$(MACH64))
537 CTF_FLAGS             = $(CTF_FLAGS_32)
539 #
540 # Flags used with genoffsets
541 #
542 GENOFFSETS_FLAGS = $(CALLSYMS)
544 OFFSETS_CREATE = $(GENOFFSETS) -s $(CTFSTABS) -r $(CTFCONVERT) \
545 $(CW) --noecho $(CW_CC_COMPILERS) -- $(GENOFFSETS_FLAGS) \
546 $(CFLAGS) $(CPPFLAGS)
548 OFFSETS_CREATE64 = $(GENOFFSETS) -s $(CTFSTABS) -r $(CTFCONVERT) \
549 $(CW) --noecho $(CW_CC_COMPILERS) -- $(GENOFFSETS_FLAGS) \
550 $(CFLAGS64) $(CPPFLAGS)
552 #
553 # tradeoff time for space (smaller is better)
554 #
555 sparc_SPACEFLAG      = -xspace -W0,-It
556 sparcv9_SPACEFLAG   = -xspace -W0,-It
557 i386_SPACEFLAG       = -xspace
558 amd64_SPACEFLAG      =
560 SPACEFLAG            = $(MACH)_SPACEFLAG
561 SPACEFLAG64         = $(MACH64)_SPACEFLAG
563 #
564 # The Sun Studio 11 compiler has changed the behaviour of integer
565 # wrap arounds and so a flag is needed to use the legacy behaviour
566 # (without this flag panics/hangs could be exposed within the source).
567 #
568 sparc_IROPTFLAG      = -W2,-xwrap_int
569 sparcv9_IROPTFLAG    = -W2,-xwrap_int
570 i386_IROPTFLAG       =
571 amd64_IROPTFLAG      =
573 IROPTFLAG           = $(MACH)_IROPTFLAG
574 IROPTFLAG64        = $(MACH64)_IROPTFLAG
576 sparc_XREGSFLAG      = -xregs=no%appl
577 sparcv9_XREGSFLAG    = -xregs=no%appl
578 i386_XREGSFLAG       =
579 amd64_XREGSFLAG      =
581 XREGSFLAG           = $(MACH)_XREGSFLAG

```

```

582 XREGSFLAG64        = $(MACH64)_XREGSFLAG
584 # dmake SOURCEDEBUG=yes ... enables source-level debugging information, and
585 # avoids stripping it.
586 SOURCEDEBUG        = $(POUND_SIGN)
587 SRCDBGBLD          = $(SOURCEDEBUG:yes=)
589 #
590 # These variables are intended ONLY for use by developers to safely pass extra
591 # flags to the compilers without unintentionally overriding Makefile-set
592 # flags. They should NEVER be set to any value in a Makefile.
593 #
594 # They come last in the associated FLAGS variable such that they can
595 # explicitly override things if necessary, there are gaps in this, but it's
596 # the best we can manage.
597 #
598 CUSERFLAGS          =
599 CUSERFLAGS64        = $(CUSERFLAGS)
600 CCUSERFLAGS         =
601 CCUSERFLAGS64       = $(CCUSERFLAGS)
603 CSOURCEDEBUGFLAGS  =
604 CCSOURCEDEBUGFLAGS =
605 $(SRCDBGBLD)CSOURCEDEBUGFLAGS = $(CCGDEBUG) -xs
606 $(SRCDBGBLD)CCSOURCEDEBUGFLAGS = $(CCGDEBUG) -xs
608 CFLAGS=             $(COPTFLAG) $(MACH)_CFLAGS $(SPACEFLAG) $(CCMODE) \
609 $(ILDOFF) $(CERRWARN) $(CSTD) $(CCUNBOUND) $(IROPTFLAG) \
610 $(CGLOBALSTATIC) $(CCNOAUTOINLINE) $(CCNOREORDER) \
611 $(CCNOAGGRESSIVELOOPS) \
612 $(CSOURCEDEBUGFLAGS) $(CUSERFLAGS)
613 CFLAGS64=          $(COPTFLAG64) $(MACH64)_CFLAGS $(SPACEFLAG64) $(CCMODE64) \
614 $(ILDOFF) $(CERRWARN) $(CSTD) $(CCUNBOUND) $(IROPTFLAG64) \
615 $(CGLOBALSTATIC) $(CCNOAUTOINLINE) $(CCNOREORDER) \
616 $(CCNOAGGRESSIVELOOPS) \
617 $(CSOURCEDEBUGFLAGS) $(CUSERFLAGS64)
618 #
619 # Flags that are used to build parts of the code that are subsequently
620 # run on the build machine (also known as the NATIVE_BUILD).
621 #
622 NATIVE_CFLAGS=     $(COPTFLAG) $(NATIVE_MACH)_CFLAGS $(CCMODE) \
623 $(ILDOFF) $(CERRWARN) $(CSTD) $(NATIVE_MACH)_CCUNBOUND) \
624 $(IROPTFLAG) $(CGLOBALSTATIC) $(CCNOAUTOINLINE) \
625 $(CCNOREORDER) $(CSOURCEDEBUGFLAGS) $(CUSERFLAGS)
627 DTEXTDOM=-DTEXT_DOMAIN="\$(TEXT_DOMAIN)" # For messaging.
628 DTS_ERRNO=-D_TS_ERRNO
629 CPPFLAGS.first= # Please keep empty. Only lower makefiles should set this.
630 CPPFLAGS.master=$(DTEXTDOM) $(DTS_ERRNO) \
631 $(ENVCPPFLAGS1) $(ENVCPPFLAGS2) $(ENVCPPFLAGS3) $(ENVCPPFLAGS4) \
632 $(ADJUNCT_PROTO:%=-I%/usr/include)
633 CPPFLAGS.native=$(ENVCPPFLAGS1) $(ENVCPPFLAGS2) $(ENVCPPFLAGS3) \
634 $(ENVCPPFLAGS4) -I$(NATIVE_ADJUNCT)/include
635 CPPFLAGS=         $(CPPFLAGS.first) $(CPPFLAGS.master)
636 AS_CPPFLAGS=      $(CPPFLAGS.first) $(CPPFLAGS.master)
637 JAVAFLAGS=        -source 1.6 -target 1.6 -Xlint:deprecation,-options
639 #
640 # For source message catalogue
641 #
642 .SUFFIXES: $(SUFFIXES) .i .po
643 MSGROOT= $(ROOT)/catalog
644 MSGDOMAIN= $(MSGROOT)/$(TEXT_DOMAIN)
645 MSGDOMAINPOFILE = $(MSGDOMAIN)/$(POFILE)
646 DCMSGDOMAIN= $(MSGROOT)/LC_TIME/$(TEXT_DOMAIN)
647 DCMSGDOMAINPOFILE = $(DCMSGDOMAIN)/$(DCFILE:.dc=.po)

```

```

649 CLOBBERFILES += $(POFILE) $(POFILES)
650 COMPILE.cpp= $(CC) -E -C $(CFLAGS) $(CPPFLAGS)
651 XGETTEXT= /usr/bin/xgettext
652 XGETTEXTFLAGS= -c TRANSLATION_NOTE
653 GNUXGETTEXT= /usr/gnu/bin/xgettext
654 GNUXGETTEXTFLAGS= --add-comments=TRANSLATION_NOTE --keyword=_ \
655 --strict --no-location --omit-header
656 BUILD.po= $(XGETTEXT) $(XGETTEXTFLAGS) -d $(<F) $<.i ;\
657 $(RM) $@ ;\
658 $(SED) "/^domain/d" < $(<F).po > $@ ;\
659 $(RM) $(<F).po $<.i

661 #
662 # This is overwritten by local Makefile when PROG is a list.
663 #
664 POFILE= $(PROG).po

666 sparc_CCFLAGS= -cg92 -compat=4 \
667 -Ooption ccfe -messages=no%anachronism \
668 $(CCERRWARN)
669 sparcv9_CCFLAGS= $(sparcv9_XARCH) -dalign -compat=5 \
670 -Ooption ccfe -messages=no%anachronism \
671 -Ooption ccfe -features=no%conststrings \
672 $(CCCREGSYM) \
673 $(CCERRWARN)
674 i386_CCFLAGS= -compat=4 \
675 -Ooption ccfe -messages=no%anachronism \
676 -Ooption ccfe -features=no%conststrings \
677 $(CCERRWARN)
678 amd64_CCFLAGS= $(amd64_XARCH) -compat=5 \
679 -Ooption ccfe -messages=no%anachronism \
680 -Ooption ccfe -features=no%conststrings \
681 $(CCERRWARN)

683 sparc_CCOPTFLAG= -O
684 sparcv9_CCOPTFLAG= -O
685 i386_CCOPTFLAG= -O
686 amd64_CCOPTFLAG= -O

688 CCOPTFLAG= $($ (MACH)_CCOPTFLAG)
689 CCOPTFLAG64= $($ (MACH64)_CCOPTFLAG)
690 CCFLAGS= $(CCOPTFLAG) $($ (MACH)_CCFLAGS) $(CCSOURCEDEBUGFLAGS) \
691 $(CCUSERFLAGS)
692 CCFLAGS64= $(CCOPTFLAG64) $($ (MACH64)_CCFLAGS) $(CCSOURCEDEBUGFLAGS) \
693 $(CCUSERFLAGS64)

695 #
696 #
697 #
698 ELFWRAP_FLAGS =
699 ELFWRAP_FLAGS64 = -64

701 #
702 # Various mapfiles that are used throughout the build, and delivered to
703 # /usr/lib/ld.
704 #
705 MAPFILE.NED_i386 = $(SRC)/common/mapfiles/common/map.noexdata
706 MAPFILE.NED_sparc =
707 MAPFILE.NED = $(MAPFILE.NED_$(MACH))
708 MAPFILE.PGA = $(SRC)/common/mapfiles/common/map.pagealign
709 MAPFILE.NES = $(SRC)/common/mapfiles/common/map.noexstk
710 MAPFILE.FLT = $(SRC)/common/mapfiles/common/map.filter
711 MAPFILE.LEX = $(SRC)/common/mapfiles/common/map.lex.yy

713 #

```

```

714 # Generated mapfiles that are compiler specific, and used throughout the
715 # build. These mapfiles are not delivered in /usr/lib/ld.
716 #
717 MAPFILE.NGB_sparc= $(SRC)/common/mapfiles/gen/sparc_cc_map.noexglobs
718 $(__GNUC64)MAPFILE.NGB_sparc= \
719 $(SRC)/common/mapfiles/gen/sparc_cc_map.noexglobs
720 MAPFILE.NGB_sparcv9= $(SRC)/common/mapfiles/gen/sparcv9_cc_map.noexglobs
721 $(__GNUC64)MAPFILE.NGB_sparcv9= \
722 $(SRC)/common/mapfiles/gen/sparcv9_cc_map.noexglobs
723 MAPFILE.NGB_i386= $(SRC)/common/mapfiles/gen/i386_cc_map.noexglobs
724 $(__GNUC64)MAPFILE.NGB_i386= \
725 $(SRC)/common/mapfiles/gen/i386_cc_map.noexglobs
726 MAPFILE.NGB_amd64= $(SRC)/common/mapfiles/gen/amd64_cc_map.noexglobs
727 $(__GNUC64)MAPFILE.NGB_amd64= \
728 $(SRC)/common/mapfiles/gen/amd64_cc_map.noexglobs
729 MAPFILE.NGB = $(MAPFILE.NGB_$(MACH))

731 #
732 # A generic interface mapfile name, used by various dynamic objects to define
733 # the interfaces and interposers the object must export.
734 #
735 MAPFILE.INT = mapfile-intf

737 #
738 # LDLIBS32 and LDLIBS64 can be set in the environment to override the following
739 # assignments.
740 #
741 # These environment settings make sure that no libraries are searched outside
742 # of the local workspace proto area:
743 # LDLIBS32=-YP,$ROOT/lib:$ROOT/usr/lib
744 # LDLIBS64=-YP,$ROOT/lib:$MACH64:$ROOT/usr/lib:$MACH64
745 #
746 LDLIBS32 = $(ENVLDLIBS1) $(ENVLDLIBS2) $(ENVLDLIBS3)
747 LDLIBS32 += $(ADJUNCT_PROTO:%=-L%/usr/lib -L%/lib)
748 LDLIBS.cmd = $(LDLIBS32)
749 LDLIBS.lib = $(LDLIBS32)

751 LDLIBS64 = $(ENVLDLIBS1:%=%/$(MACH64)) \
752 $(ENVLDLIBS2:%=%/$(MACH64)) \
753 $(ENVLDLIBS3:%=%/$(MACH64))
754 LDLIBS64 += $(ADJUNCT_PROTO:%=-L%/usr/lib/$(MACH64) -L%/lib/$(MACH64))

756 #
757 # Define compilation macros.
758 #
759 COMPILE.c= $(CC) $(CFLAGS) $(CPPFLAGS) -c
760 COMPILE64.c= $(CC) $(CFLAGS64) $(CPPFLAGS) -c
761 COMPILE.cc= $(CCC) $(CCFLAGS) $(CPPFLAGS) -c
762 COMPILE64.ccc= $(CCC) $(CCFLAGS64) $(CPPFLAGS) -c
763 COMPILE.s= $(AS) $(ASFLAGS) $(AS_CPPFLAGS)
764 COMPILE64.s= $(AS) $(ASFLAGS) $($ (MACH64)_AS_XARCH) $(AS_CPPFLAGS)
765 COMPILE.d= $(DTRACE) -G -32
766 COMPILE64.d= $(DTRACE) -G -64
767 COMPILE.b= $(ELFWRAP) $(ELFWRAP_FLAGS$(CLASS))
768 COMPILE64.b= $(ELFWRAP) $(ELFWRAP_FLAGS$(CLASS))

770 CLASSPATH= .
771 COMPILE.java= $(JAVAC) $(JAVAFLAGS) -classpath $(CLASSPATH)

773 #
774 # Link time macros
775 #
776 CCNEEDED = -lc
777 CCEXTNEEDED = -lcrun -lcstd
778 $(__GNUC)CCNEEDED = -L$(GCCLIBDIR) -lstc++ -lgcc_s
779 $(__GNUC)CCEXTNEEDED = $(CCNEEDED)

```

```

781 LINK.c=          $(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS)
782 LINK64.c=        $(CC) $(CFLAGS64) $(CPPFLAGS) $(LDFLAGS)
783 NORUNPATH=        -norunpath -nolib
784 LINK.cc=          $(CCC) $(CCFLAGS) $(CPPFLAGS) $(NORUNPATH) \
785                  $(LDFLAGS) $(CCNEEDED)
786 LINK64.cc=        $(CCC) $(CCFLAGS64) $(CPPFLAGS) $(NORUNPATH) \
787                  $(LDFLAGS) $(CCNEEDED)

789 #
790 # lint macros
791 #
792 # Note that the undefine of __PRAGMA_REDEFINE_EXTNAME can be removed once
793 # ON is built with a version of lint that has the fix for 4484186.
794 #
795 ALWAYS_LINT_DEFS = -errtags=yes -s
796 ALWAYS_LINT_DEFS += -erroff=E_PTRDIFF_OVERFLOW
797 ALWAYS_LINT_DEFS += -erroff=E_ASSIGN_NARROW_CONV
798 ALWAYS_LINT_DEFS += -U__PRAGMA_REDEFINE_EXTNAME
799 ALWAYS_LINT_DEFS += $(C99LMODE)
800 ALWAYS_LINT_DEFS += -errsecurity=$(SECLEVEL)
801 ALWAYS_LINT_DEFS += -erroff=E_SEC_CREAT_WITHOUT_EXCL
802 ALWAYS_LINT_DEFS += -erroff=E_SEC_FORBIDDEN_WARN_CREAT
803 # XX64 -- really only needed for amd64 lint
804 ALWAYS_LINT_DEFS += -erroff=E_ASSIGN_INT_TO_SMALL_INT
805 ALWAYS_LINT_DEFS += -erroff=E_CAST_INT_CONST_TO_SMALL_INT
806 ALWAYS_LINT_DEFS += -erroff=E_CAST_INT_TO_SMALL_INT
807 ALWAYS_LINT_DEFS += -erroff=E_CAST_TO_PTR_FROM_INT
808 ALWAYS_LINT_DEFS += -erroff=E_COMP_INT_WITH_LARGE_INT
809 ALWAYS_LINT_DEFS += -erroff=E_INTEGRAL_CONST_EXP_EXPECTED
810 ALWAYS_LINT_DEFS += -erroff=E_PASS_INT_TO_SMALL_INT
811 ALWAYS_LINT_DEFS += -erroff=E_PTR_CONV_LOSES_BITS

813 # This forces lint to pick up note.h and sys/note.h from Devpro rather than
814 # from the proto area. The note.h that ON delivers would disable NOTE().
815 ONLY_LINT_DEFS = -I$(SPRO_VROOT)/prod/include/lint

817 SECLEVEL=         core
818 LINT.c=            $(LINT) $(ONLY_LINT_DEFS) $(LINTFLAGS) $(CPPFLAGS) \
819                  $(ALWAYS_LINT_DEFS)
820 LINT64.c=          $(LINT) $(ONLY_LINT_DEFS) $(LINTFLAGS64) $(CPPFLAGS) \
821                  $(ALWAYS_LINT_DEFS)
822 LINT.s=            $(LINT.c)

824 # For some future builds, NATIVE_MACH and MACH might be different.
825 # Therefore, NATIVE_MACH needs to be redefined in the
826 # environment as 'uname -p' to override this macro.
827 #
828 # For now at least, we cross-compile amd64 on i386 machines.
829 NATIVE_MACH=       $(MACH:amd64=i386)

831 # Define native compilation macros
832 #

834 # Base directory where compilers are loaded.
835 # Defined here so it can be overridden by developer.
836 #
837 SPRO_ROOT=         $(BUILD_TOOLS)/SUNwsprio
838 SPRO_VROOT=        $(SPRO_ROOT)/SS12
839 GNU_ROOT=          /usr

841 $(__GNUC)PRIMARY_CC= gcc4,$(GNUC_ROOT)/bin/gcc.gnu
842 $(__SUNC)PRIMARY_CC= studio12,$(SPRO_VROOT)/bin/cc.sun
843 $(__GNUC)PRIMARY_CCC= gcc4,$(GNUC_ROOT)/bin/g++.gnu
844 $(__SUNC)PRIMARY_CCC= studio12,$(SPRO_VROOT)/bin/CC.sun

```

```

846 CW_CC_COMPILERS= $(PRIMARY_CC:%--primary %) $(SHADOW_CCS:%--shadow %)
847 CW_CCC_COMPILERS= $(PRIMARY_CCC:%--primary %) $(SHADOW_CCCS:%--shadow %)

850 # Till SS12ul formally becomes the NV CBE, LINT is hard
851 # coded to be picked up from the $SPRO_ROOT/sunstudio12.1/
852 # location. Impacted variables are sparc_LINT, sparcv9_LINT,
853 # i386_LINT, amd64_LINT.
854 # Reset them when SS12ul is rolled out.
855 #

857 # Specify platform compiler versions for languages
858 # that we use (currently only c and c++).
859 #
860 CW=                $(ONBLD_TOOLS)/bin/$(MACH)/cw

862 BUILD_CC=          $(CW) $(CW_CC_COMPILERS) --
863 BUILD_CCC=         $(CW) -C $(CW_CCC_COMPILERS) --
864 BUILD_CPP=         /usr/ccs/lib/cpp
865 BUILD_LD=          /usr/ccs/bin/ld
866 BUILD_LINT=        $(SPRO_ROOT)/sunstudio12.1/bin/lint

868 $(MACH)_CC=        $(BUILD_CC)
869 $(MACH)_CCC=       $(BUILD_CCC)
870 $(MACH)_CPP=       $(BUILD_CPP)
871 $(MACH)_LD=        $(BUILD_LD)
872 $(MACH)_LINT=     $(BUILD_LINT)
873 $(MACH64)_CC=     $(BUILD_CC)
874 $(MACH64)_CCC=    $(BUILD_CCC)
875 $(MACH64)_CPP=    $(BUILD_CPP)
876 $(MACH64)_LD=    $(BUILD_LD)
877 $(MACH64)_LINT=   $(BUILD_LINT)

879 sparc_AS=          /usr/ccs/bin/as -xregsym=no
880 sparcv9_AS=        $(MACH)_AS

882 i386_AS=           /usr/ccs/bin/as
883 $(__GNUC)i386_AS= $(ONBLD_TOOLS)/bin/$(MACH)/aw
884 amd64_AS=          $(ONBLD_TOOLS)/bin/$(MACH)/aw

886 NATIVECC=          $(MACH)_CC
887 NATIVECCC=         $(MACH)_CCC
888 NATIVECPP=         $(MACH)_CPP
889 NATIVEAS=          $(MACH)_AS
890 NATIVELD=          $(MACH)_LD
891 NATIVELINT=        $(MACH)_LINT

893 #
894 # Makefile.master.64 overrides these settings
895 #
896 CC=                $(NATIVECC)
897 CCC=               $(NATIVECCC)
898 CPP=               $(NATIVECPP)
899 AS=                $(NATIVEAS)
900 LD=                $(NATIVELD)
901 LINT=              $(NATIVELINT)

903 # Pass -Y flag to cpp (method of which is release-dependent)
904 CCYFLAG=           -Y I,

906 BDIRECT=           -Bdirect
907 BYNAMIC=           -Bdynamic
908 BLOCAL=            -Blocal
909 BNODIRECT=         -Bnodirect
910 BREDUCE=           -Breduce
911 BSTATIC=           -Bstatic

```

```

913 ZDEFS=          -zdefs
914 ZDIRECT=        -zdirect
915 ZIGNORE=        -zignore
916 ZINITFIRST=     -zinitfirst
917 ZINTERPOSE=     -zinterpose
918 ZLAZYLOAD=      -zlazyload
919 ZLOADFLTR=      -zloadfltr
920 ZMULDEFS=       -zmuldefs
921 ZNODEFAULTLIB=  -znodefaultlib
922 ZNODEFS=        -znodefs
923 ZNODELETE=      -znodelete
924 ZNODLOPEN=      -znodlopen
925 ZNODUMP=        -znodump
926 ZNOLAZYLOAD=    -znolazyload
927 ZNOLDYNSYM=     -znoldynsym
928 ZNORELOC=       -znoreloc
929 ZNOVERSION=     -znoversion
930 ZRECORD=        -zrecord
931 ZREDLOCSYM=     -zredlocsymb
932 ZTEXT=          -ztext
933 ZVERBOSE=       -zverbose

935 GSHARED=        -G
936 CCMT=           -mt

938 # Handle different PIC models on different ISAs
939 # (May be overridden by lower-level Makefiles)

941 sparc_C_PICFLAGS = -fpic
942 sparcv9_C_PICFLAGS = -fpic
943 i386_C_PICFLAGS = -fpic
944 amd64_C_PICFLAGS = -fpic
945 C_PICFLAGS = $(($(MACH)_C_PICFLAGS))
946 C_PICFLAGS64 = $(($(MACH64)_C_PICFLAGS))

948 sparc_C_BIGPICFLAGS = -fPIC
949 sparcv9_C_BIGPICFLAGS = -fPIC
950 i386_C_BIGPICFLAGS = -fPIC
951 amd64_C_BIGPICFLAGS = -fPIC
952 C_BIGPICFLAGS = $(($(MACH)_C_BIGPICFLAGS))
953 C_BIGPICFLAGS64 = $(($(MACH64)_C_BIGPICFLAGS))

955 # CC requires there to be no space between '-K' and 'pic' or 'PIC'.
956 # and does not support -f
957 sparc_CC_PICFLAGS = -_cc=-Kpic -_gcc=-fpic
958 sparcv9_CC_PICFLAGS = -_cc=-KPIC -_gcc=-fPIC
959 i386_CC_PICFLAGS = -_cc=-Kpic -_gcc=-fpic
960 amd64_CC_PICFLAGS = -_cc=-KPIC -_gcc=-fpic
961 CC_PICFLAGS = $(($(MACH)_CC_PICFLAGS))
962 CC_PICFLAGS64 = $(($(MACH64)_CC_PICFLAGS))

964 AS_PICFLAGS=     -K pic
965 AS_BIGPICFLAGS= -K PIC

967 #
968 # Default label for CTF sections
969 #
970 CTFCVTFLAGS=     -L VERSION
970 CTFCVTFLAGS=     -i -L VERSION

972 #
973 # Override to pass module-specific flags to ctfmerge. Currently used only by
974 # krtld to turn on fuzzy matching, and source-level debugging to inhibit
975 # stripping.
976 #

```

```

977 CTFMRGFLAGS=

979 #
980 # Make the transition between old and new CTF Tools. The new ctf tools
981 # do not support stabs (eg. Sun Studio). By setting BUILD_OLD_CTF_TOOLS
982 # here or in the environment file, the old ones will be built.
983 #
984 BUILD_NEW_CTF_TOOLS=
985 BUILD_OLD_CTF_TOOLS=$(POUND_SIGN)
986 $(BUILD_OLD_CTF_TOOLS)BUILD_NEW_CTF_TOOLS= $(POUND_SIGN)

988 CTFCONVERT_O      = $(CTFCONVERT) $(CTFCVTFLAGS) $@

990 # Rules (normally from make.rules) and macros which are used for post
991 # processing files. Normally, these do stripping of the comment section
992 # automatically.
993 # RELEASE_CM: Should be edited to reflect the release.
994 # POST_PROCESS_O: Post-processing for '.o' files (typically C source)
995 # POST_PROCESS_S_O: Post-processing for '.o' files built from assembly
996 # POST_PROCESS_CC_O: Post-processing for '.o' files built from C++
997 # RELEASE_CM: Should be edited to reflect the release.
998 # POST_PROCESS_O: Post-processing for '.o' files.
999 # POST_PROCESS_A: Post-processing for '.a' files (currently null).
1000 # POST_PROCESS_SO: Post-processing for '.so' files.
1001 # POST_PROCESS: Post-processing for executable files (no suffix).
1002 #
1003 # Note that these macros are not completely generalized as they are to be
1004 # used with the file name to be processed following.
1005 #
1006 # It is left as an exercise to Release Engineering to embellish the generation
1007 # of the release comment string.
1008 #
1009 # If this is a standard development build:
1010 # compress the comment section (mcs -c)
1011 # add the standard comment (mcs -a $(RELEASE_CM))
1012 # add the development specific comment (mcs -a $(DEV_CM))
1013 #
1014 # If this is an installation build:
1015 # delete the comment section (mcs -d)
1016 # add the standard comment (mcs -a $(RELEASE_CM))
1017 # add the development specific comment (mcs -a $(DEV_CM))
1018 #
1019 # If this is an release build:
1020 # delete the comment section (mcs -d)
1021 # add the standard comment (mcs -a $(RELEASE_CM))
1022 #
1023 # The following list of macros are used in the definition of RELEASE_CM
1024 # which is used to label all binaries in the build:
1025 #
1026 # RELEASE Specific release of the build, eg: 5.2
1027 # RELEASE_MAJOR Major version number part of $(RELEASE)
1028 # RELEASE_MINOR Minor version number part of $(RELEASE)
1029 # VERSION Version of the build (alpha, beta, Generic)
1030 # PATCHID If this is a patch this value should contain
1031 # the patchid value (eg: "Generic 100832-01"), otherwise
1032 # it will be set to $(VERSION)
1033 # RELEASE_DATE Date of the Release Build
1034 # PATCH_DATE Date the patch was created, if this is blank it
1035 # will default to the RELEASE_DATE
1036 #
1037 #
1038 #
1039 #
1040 #
1041 #
1042 #
1043 #
1044 #
1045 #
1046 #
1047 #
1048 #
1049 #
1050 #
1051 #
1052 #
1053 #
1054 #
1055 #
1056 #
1057 #
1058 #
1059 #
1060 #
1061 #
1062 #
1063 #
1064 #
1065 #
1066 #
1067 #
1068 #
1069 #
1070 #
1071 #
1072 #
1073 #
1074 #
1075 #
1076 #
1077 #
1078 #
1079 #
1080 #
1081 #
1082 #
1083 #
1084 #
1085 #
1086 #
1087 #
1088 #
1089 #
1090 #
1091 #
1092 #
1093 #
1094 #
1095 #
1096 #
1097 #
1098 #
1099 #
1100 #
1101 #
1102 #
1103 #
1104 #
1105 #
1106 #
1107 #
1108 #
1109 #
1110 #
1111 #
1112 #
1113 #
1114 #
1115 #
1116 #
1117 #
1118 #
1119 #
1120 #
1121 #
1122 #
1123 #
1124 #
1125 #
1126 #
1127 #
1128 #
1129 #
1130 #
1131 #
1132 #
1133 #
1134 #
1135 #
1136 #
1137 #
1138 #
1139 #
1140 #
1141 #
1142 #
1143 #
1144 #
1145 #
1146 #
1147 #
1148 #
1149 #
1150 #
1151 #
1152 #
1153 #
1154 #
1155 #
1156 #
1157 #
1158 #
1159 #
1160 #
1161 #
1162 #
1163 #
1164 #
1165 #
1166 #
1167 #
1168 #
1169 #
1170 #
1171 #
1172 #
1173 #
1174 #
1175 #
1176 #
1177 #
1178 #
1179 #
1180 #
1181 #
1182 #
1183 #
1184 #
1185 #
1186 #
1187 #
1188 #
1189 #
1190 #
1191 #
1192 #
1193 #
1194 #
1195 #
1196 #
1197 #
1198 #
1199 #
1200 #
1201 #
1202 #
1203 #
1204 #
1205 #
1206 #
1207 #
1208 #
1209 #
1210 #
1211 #
1212 #
1213 #
1214 #
1215 #
1216 #
1217 #
1218 #
1219 #
1220 #
1221 #
1222 #
1223 #
1224 #
1225 #
1226 #
1227 #
1228 #
1229 #
1230 #
1231 #
1232 #
1233 #
1234 #
1235 #
1236 #
1237 #
1238 #
1239 #
1240 #
1241 #
1242 #
1243 #
1244 #
1245 #
1246 #
1247 #
1248 #
1249 #
1250 #
1251 #
1252 #
1253 #
1254 #
1255 #
1256 #
1257 #
1258 #
1259 #
1260 #
1261 #
1262 #
1263 #
1264 #
1265 #
1266 #
1267 #
1268 #
1269 #
1270 #
1271 #
1272 #
1273 #
1274 #
1275 #
1276 #
1277 #
1278 #
1279 #
1280 #
1281 #
1282 #
1283 #
1284 #
1285 #
1286 #
1287 #
1288 #
1289 #
1290 #
1291 #
1292 #
1293 #
1294 #
1295 #
1296 #
1297 #
1298 #
1299 #
1300 #
1301 #
1302 #
1303 #
1304 #
1305 #
1306 #
1307 #
1308 #
1309 #
1310 #
1311 #
1312 #
1313 #
1314 #
1315 #
1316 #
1317 #
1318 #
1319 #
1320 #
1321 #
1322 #
1323 #
1324 #
1325 #
1326 #
1327 #
1328 #
1329 #
1330 #
1331 #
1332 #
1333 #
1334 #
1335 #
1336 #
1337 #
1338 #
1339 #
1340 #
1341 #
1342 #
1343 #
1344 #
1345 #
1346 #
1347 #
1348 #
1349 #
1350 #
1351 #
1352 #
1353 #
1354 #
1355 #
1356 #
1357 #
1358 #
1359 #
1360 #
1361 #
1362 #
1363 #
1364 #
1365 #
1366 #
1367 #
1368 #
1369 #
1370 #
1371 #
1372 #
1373 #
1374 #
1375 #
1376 #
1377 #
1378 #
1379 #
1380 #
1381 #
1382 #
1383 #
1384 #
1385 #
1386 #
1387 #
1388 #
1389 #
1390 #
1391 #
1392 #
1393 #
1394 #
1395 #
1396 #
1397 #
1398 #
1399 #
1400 #
1401 #
1402 #
1403 #
1404 #
1405 #
1406 #
1407 #
1408 #
1409 #
1410 #
1411 #
1412 #
1413 #
1414 #
1415 #
1416 #
1417 #
1418 #
1419 #
1420 #
1421 #
1422 #
1423 #
1424 #
1425 #
1426 #
1427 #
1428 #
1429 #
1430 #
1431 #
1432 #
1433 #
1434 #
1435 #
1436 #
1437 #
1438 #
1439 #
1440 #
1441 #
1442 #
1443 #
1444 #
1445 #
1446 #
1447 #
1448 #
1449 #
1450 #
1451 #
1452 #
1453 #
1454 #
1455 #
1456 #
1457 #
1458 #
1459 #
1460 #
1461 #
1462 #
1463 #
1464 #
1465 #
1466 #
1467 #
1468 #
1469 #
1470 #
1471 #
1472 #
1473 #
1474 #
1475 #
1476 #
1477 #
1478 #
1479 #
1480 #
1481 #
1482 #
1483 #
1484 #
1485 #
1486 #
1487 #
1488 #
1489 #
1490 #
1491 #
1492 #
1493 #
1494 #
1495 #
1496 #
1497 #
1498 #
1499 #
1500 #
1501 #
1502 #
1503 #
1504 #
1505 #
1506 #
1507 #
1508 #
1509 #
1510 #
1511 #
1512 #
1513 #
1514 #
1515 #
1516 #
1517 #
1518 #
1519 #
1520 #
1521 #
1522 #
1523 #
1524 #
1525 #
1526 #
1527 #
1528 #
1529 #
1530 #
1531 #
1532 #
1533 #
1534 #
1535 #
1536 #
1537 #
1538 #
1539 #
1540 #
1541 #
1542 #
1543 #
1544 #
1545 #
1546 #
1547 #
1548 #
1549 #
1550 #
1551 #
1552 #
1553 #
1554 #
1555 #
1556 #
1557 #
1558 #
1559 #
1560 #
1561 #
1562 #
1563 #
1564 #
1565 #
1566 #
1567 #
1568 #
1569 #
1570 #
1571 #
1572 #
1573 #
1574 #
1575 #
1576 #
1577 #
1578 #
1579 #
1580 #
1581 #
1582 #
1583 #
1584 #
1585 #
1586 #
1587 #
1588 #
1589 #
1590 #
1591 #
1592 #
1593 #
1594 #
1595 #
1596 #
1597 #
1598 #
1599 #
1600 #
1601 #
1602 #
1603 #
1604 #
1605 #
1606 #
1607 #
1608 #
1609 #
1610 #
1611 #
1612 #
1613 #
1614 #
1615 #
1616 #
1617 #
1618 #
1619 #
1620 #
1621 #
1622 #
1623 #
1624 #
1625 #
1626 #
1627 #
1628 #
1629 #
1630 #
1631 #
1632 #
1633 #
1634 #
1635 #
1636 #
1637 #
1638 #
1639 #
1640 #
1641 #
1642 #
1643 #
1644 #
1645 #
1646 #
1647 #
1648 #
1649 #
1650 #
1651 #
1652 #
1653 #
1654 #
1655 #
1656 #
1657 #
1658 #
1659 #
1660 #
1661 #
1662 #
1663 #
1664 #
1665 #
1666 #
1667 #
1668 #
1669 #
1670 #
1671 #
1672 #
1673 #
1674 #
1675 #
1676 #
1677 #
1678 #
1679 #
1680 #
1681 #
1682 #
1683 #
1684 #
1685 #
1686 #
1687 #
1688 #
1689 #
1690 #
1691 #
1692 #
1693 #
1694 #
1695 #
1696 #
1697 #
1698 #
1699 #
1700 #
1701 #
1702 #
1703 #
1704 #
1705 #
1706 #
1707 #
1708 #
1709 #
1710 #
1711 #
1712 #
1713 #
1714 #
1715 #
1716 #
1717 #
1718 #
1719 #
1720 #
1721 #
1722 #
1723 #
1724 #
1725 #
1726 #
1727 #
1728 #
1729 #
1730 #
1731 #
1732 #
1733 #
1734 #
1735 #
1736 #
1737 #
1738 #
1739 #
1740 #
1741 #
1742 #
1743 #
1744 #
1745 #
1746 #
1747 #
1748 #
1749 #
1750 #
1751 #
1752 #
1753 #
1754 #
1755 #
1756 #
1757 #
1758 #
1759 #
1760 #
1761 #
1762 #
1763 #
1764 #
1765 #
1766 #
1767 #
1768 #
1769 #
1770 #
1771 #
1772 #
1773 #
1774 #
1775 #
1776 #
1777 #
1778 #
1779 #
1780 #
1781 #
1782 #
1783 #
1784 #
1785 #
1786 #
1787 #
1788 #
1789 #
1790 #
1791 #
1792 #
1793 #
1794 #
1795 #
1796 #
1797 #
1798 #
1799 #
1800 #
1801 #
1802 #
1803 #
1804 #
1805 #
1806 #
1807 #
1808 #
1809 #
1810 #
1811 #
1812 #
1813 #
1814 #
1815 #
1816 #
1817 #
1818 #
1819 #
1820 #
1821 #
1822 #
1823 #
1824 #
1825 #
1826 #
1827 #
1828 #
1829 #
1830 #
1831 #
1832 #
1833 #
1834 #
1835 #
1836 #
1837 #
1838 #
1839 #
1840 #
1841 #
1842 #
1843 #
1844 #
1845 #
1846 #
1847 #
1848 #
1849 #
1850 #
1851 #
1852 #
1853 #
1854 #
1855 #
1856 #
1857 #
1858 #
1859 #
1860 #
1861 #
1862 #
1863 #
1864 #
1865 #
1866 #
1867 #
1868 #
1869 #
1870 #
1871 #
1872 #
1873 #
1874 #
1875 #
1876 #
1877 #
1878 #
1879 #
1880 #
1881 #
1882 #
1883 #
1884 #
1885 #
1886 #
1887 #
1888 #
1889 #
1890 #
1891 #
1892 #
1893 #
1894 #
1895 #
1896 #
1897 #
1898 #
1899 #
1900 #
1901 #
1902 #
1903 #
1904 #
1905 #
1906 #
1907 #
1908 #
1909 #
1910 #
1911 #
1912 #
1913 #
1914 #
1915 #
1916 #
1917 #
1918 #
1919 #
1920 #
1921 #
1922 #
1923 #
1924 #
1925 #
1926 #
1927 #
1928 #
1929 #
1930 #
1931 #
1932 #
1933 #
1934 #
1935 #
1936 #
1937 #
1938 #
1939 #
1940 #
1941 #
1942 #
1943 #
1944 #
1945 #
1946 #
1947 #
1948 #
1949 #
1950 #
1951 #
1952 #
1953 #
1954 #
1955 #
1956 #
1957 #
1958 #
1959 #
1960 #
1961 #
1962 #
1963 #
1964 #
1965 #
1966 #
1967 #
1968 #
1969 #
1970 #
1971 #
1972 #
1973 #
1974 #
1975 #
1976 #
1977 #
1978 #
1979 #
1980 #
1981 #
1982 #
1983 #
1984 #
1985 #
1986 #
1987 #
1988 #
1989 #
1990 #
1991 #
1992 #
1993 #
1994 #
1995 #
1996 #
1997 #
1998 #
1999 #
2000 #

```



```

1041 PATCH_DATE=      $(RELEASE_DATE)
1042 RELEASE_CM=      "@$(POUND_SIGN)SunOS $(RELEASE) $(PATCHID) $(PATCH_DATE)"
1043 DEV_CM=          "@$(POUND_SIGN)SunOS Internal Development: non-nightly build"

1045 PROCESS_COMMENT= @?${MCS} -d -a $(RELEASE_CM) -a $(DEV_CM)
1046 $(RELEASE_BUILD)PROCESS_COMMENT= @?${MCS} -d -a $(RELEASE_CM)

1048 STRIP_STABS=     $(STRIP) -x $@
1049 $(SRCDBGBLD)STRIP_STABS=
        :

1051 POST_PROCESS_O=
1052 POST_PROCESS_S_O=
1053 POST_PROCESS_CC_O=
1054 POST_PROCESS_A=
1055 POST_PROCESS_SO=  $(PROCESS_COMMENT) $@ ; $(STRIP_STABS) ; \
1056                  $(ELFSIGN_OBJECT)
1057 POST_PROCESS=     $(PROCESS_COMMENT) $@ ; $(STRIP_STABS) ; \
1058                  $(ELFSIGN_OBJECT)

1060 #
1061 # chk4ubin is a tool that inspects a module for a symbol table
1062 # ELF section size which can trigger an OBP bug on older platforms.
1063 # This problem affects only specific sun4u bootable modules.
1064 #
1065 CHK4UBIN=         $(ONBLD_TOOLS)/bin/$(MACH)/chk4ubin
1066 CHK4UBINFLAGS=
1067 CHK4UBINARY=     $(CHK4UBIN) $(CHK4UBINFLAGS) $@

1069 #
1070 # PKGARCHIVE specifies the default location where packages should be
1071 # placed if built.
1072 #
1073 $(RELEASE_BUILD)PKGARCHIVESUFFIX= -nd
1074 PKGARCHIVE=$(SRC)/.../packages/$(MACH)/nightly$(PKGARCHIVESUFFIX)

1076 #
1077 # The repositories will be created with these publisher settings. To
1078 # update an image to the resulting repositories, this must match the
1079 # publisher name provided to "pkg set-publisher."
1080 #
1081 PKGPUBLISHER_REDIST= on-nightly
1082 PKGPUBLISHER_NONREDIST= on-extra

1084 #          Default build rules which perform comment section post-processing.
1085 #
1086 .c:
1087     $(LINK.c) -o $@ $< $(LDLIBS)
1088     $(POST_PROCESS)
1089 .c.o:
1090     $(COMPILE.c) $(OUTPUT_OPTION) $< $(CTFCONVERT_HOOK)
1091     $(POST_PROCESS_O)
1092 .c.a:
1093     $(COMPILE.c) -o $% $<
1094     $(PROCESS_COMMENT) $%
1095     $(AR) $(ARFLAGS) $@ $%
1096     $(RM) $%
1097 .s.o:
1098     $(COMPILE.s) -o $@ $<
1099     $(POST_PROCESS_S_O)
1100     $(POST_PROCESS_O)
1101 .s.a:
1102     $(COMPILE.s) -o $% $<
1103     $(PROCESS_COMMENT) $%
1104     $(AR) $(ARFLAGS) $@ $%
1105     $(RM) $%
1106     .cc:

```

```

1106     $(LINK.cc) -o $@ $< $(LDLIBS)
1107     $(POST_PROCESS)
1108 .cc.o:
1109     $(COMPILE.cc) $(OUTPUT_OPTION) $<
1110     $(POST_PROCESS_CC_O)
1111     $(POST_PROCESS_O)
1112 .cc.a:
1113     $(COMPILE.cc) -o $% $<
1114     $(AR) $(ARFLAGS) $@ $%
1115     $(PROCESS_COMMENT) $%
1116     $(RM) $%
1117 .y:
1118     $(YACC.y) $<
1119     $(LINK.c) -o $@ y.tab.c $(LDLIBS)
1120     $(POST_PROCESS)
1121     $(RM) y.tab.c
1122 .y.o:
1123     $(YACC.y) $<
1124     $(COMPILE.c) -o $@ y.tab.c $(CTFCONVERT_HOOK)
1125     $(POST_PROCESS_O)
1126     $(RM) y.tab.c
1127 .l:
1128     $(RM) $*.c
1129     $(LEX.l) $< > $*.c
1130     $(LINK.c) -o $@ $*.c -ll $(LDLIBS)
1131     $(POST_PROCESS)
1132     $(RM) $*.c
1133 .l.o:
1134     $(RM) $*.c
1135     $(LEX.l) $< > $*.c
1136     $(COMPILE.c) -o $@ $*.c $(CTFCONVERT_HOOK)
1137     $(POST_PROCESS_O)
1138     $(RM) $*.c
1139 .bin.o:
1140     $(COMPILE.b) -o $@ $<
1141     $(POST_PROCESS_O)

1143 .java.class:
1144     $(COMPILE.java) $<

1146 # Bourne and Korn shell script message catalog build rules.
1147 # We extract all gettext strings with sed(1) (being careful to permit
1148 # multiple gettext strings on the same line), weed out the dups, and
1149 # build the catalogue with awk(1).

1151 .sh.po .ksh.po:
1152     $(SED) -n -e ":a"
1153             -e "h"
1154             -e "s/.*gettext *\([^\"]*\).*\1/p"
1155             -e "x"
1156             -e "s/\([^\"]*\)gettext *\([^\"]*\)\1/2/"
1157             -e "t a"
1158     $< | sort -u | $(AWK) '{ print "msgid\t" $$0 "\nmsgstr" }' > $@

1160 #
1161 # Python and Perl executable and message catalog build rules.
1162 #
1163 .SUFFIXES: .pl .pm .py .pyc

1165 .pl:
1166     $(RM) $@;
1167     $(SED) -e "s@TEXT_DOMAIN@\"$(TEXT_DOMAIN)\"@ $< > $@;
1168     $(CHMOD) +x $@

1170 .py:

```

new/usr/src/Makefile.master

19

```
1171 $(RM) $@; $(SED) \  
1172 -e "1s:^(#@PYTHON@:\#!$(PYSHEBANG):" \  
1173 -e "1s:^(#@TOOLS_PYTHON@:\#!$(TOOLS_PYTHON):" \  
1174 < $< > $@; $(CHMOD) +x $@  
  
1176 .py.pyc:  
1177 $(RM) $@  
1178 $(PYTHON) -mpy_compile $<  
1179 @[ $(<)c = $@ ] || $(MV) $(<)c $@  
  
1181 .py.po:  
1182 $(GNUXGETTEXT) $(GNUXGETFLAGS) -d $(<F:%.py=%) $< ;  
  
1184 .pl.po .pm.po:  
1185 $(XGETTEXT) $(XGETFLAGS) -d $(<F) $< ;  
1186 $(RM) $@ ;  
1187 $(SED) "/^domain/d" < $(<F).po > $@ ;  
1188 $(RM) $(<F).po  
  
1190 #  
1191 # When using xgettext, we want messages to go to the default domain,  
1192 # rather than the specified one. This special version of the  
1193 # COMPILER.cpp macro effectively prevents expansion of TEXT_DOMAIN,  
1194 # causing xgettext to put all messages into the default domain.  
1195 #  
1196 CPPFORPO=$(COMPILER.cpp:\ "$(TEXT_DOMAIN)"=TEXT_DOMAIN)  
  
1198 .c.i:  
1199 $(CPPFORPO) $< > $@  
  
1201 .h.i:  
1202 $(CPPFORPO) $< > $@  
  
1204 .y.i:  
1205 $(YACC) -d $<  
1206 $(CPPFORPO) y.tab.c > $@  
1207 $(RM) y.tab.c  
  
1209 .l.i:  
1210 $(LEX) $<  
1211 $(CPPFORPO) lex.yy.c > $@  
1212 $(RM) lex.yy.c  
  
1214 .c.po:  
1215 $(CPPFORPO) $< > $<.i  
1216 $(BUILD.po)  
  
1218 .cc.po:  
1219 $(CPPFORPO) $< > $<.i  
1220 $(BUILD.po)  
  
1222 .y.po:  
1223 $(YACC) -d $<  
1224 $(CPPFORPO) y.tab.c > $<.i  
1225 $(BUILD.po)  
1226 $(RM) y.tab.c  
  
1228 .l.po:  
1229 $(LEX) $<  
1230 $(CPPFORPO) lex.yy.c > $<.i  
1231 $(BUILD.po)  
1232 $(RM) lex.yy.c  
  
1234 #  
1235 # Rules to perform stylistic checks  
1236 #
```

new/usr/src/Makefile.master

20

```
1237 .SUFFIXES: .x .xml .check .xmlchk  
  
1239 .h.check:  
1240 $(DOT_H_CHECK)  
  
1242 .x.check:  
1243 $(DOT_X_CHECK)  
  
1245 .xml.xmlchk:  
1246 $(MANIFEST_CHECK)  
  
1248 #  
1249 # Include rules to render automated sccs get rules "safe".  
1250 #  
1251 include $(SRC)/Makefile.noget
```

new/usr/src/cmd/ctfconvert/ctfconvert.c

1

```
10474 Thu Apr 25 02:20:39 2019
new/usr/src/cmd/ctfconvert/ctfconvert.c
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
```

```
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 at
9  * http://www.illumos.org/license/CDDL.
10 */
12 /*
13  * Copyright (c) 2019, Joyent, Inc.
14  * Copyright (c) 2015, Joyent, Inc.
15 */
16 /*
17  * Create CTF from extant debugging information
18 */
20 #include <stdio.h>
21 #include <unistd.h>
22 #include <stdlib.h>
23 #include <stdarg.h>
24 #include <sys/types.h>
25 #include <sys/stat.h>
26 #include <fcntl.h>
27 #include <errno.h>
28 #include <libelf.h>
29 #include <libctf.h>
30 #include <string.h>
31 #include <libgen.h>
32 #include <limits.h>
33 #include <strings.h>
34 #include <sys/debug.h>
36 #define CTFCONVERT_OK          0
37 #define CTFCONVERT_FATAL      1
38 #define CTFCONVERT_USAGE      2
40 #define CTFCONVERT_DEFAULT_NTHREADS 4
42 #define CTFCONVERT_ALTEXEC    "CTFCONVERT_ALTEXEC"
44 static char *ctfconvert_progname;
46 static void
47 ctfconvert_fatal(const char *fmt, ...)
48 {
49     va_list ap;
51     (void) fprintf(stderr, "%s: ", ctfconvert_progname);
52     va_start(ap, fmt);
```

new/usr/src/cmd/ctfconvert/ctfconvert.c

2

```
53     (void) vfprintf(stderr, fmt, ap);
54     va_end(ap);
56     exit(CTFCONVERT_FATAL);
57 }
60 static void
61 ctfconvert_usage(const char *fmt, ...)
62 {
63     if (fmt != NULL) {
64         va_list ap;
66         (void) fprintf(stderr, "%s: ", ctfconvert_progname);
67         va_start(ap, fmt);
68         (void) vfprintf(stderr, fmt, ap);
69         va_end(ap);
70     }
72     (void) fprintf(stderr, "Usage: %s [-ims] [-j nthrs] [-l label | "
73 (void) fprintf(stderr, "Usage: %s [-is] [-j nthrs] [-l label | "
74 " -L labelenv] [-o outfile] input\n"
75 "\n"
76 "\t-i ignore files not built partially from C sources\n"
77 "\t-j use nthrs threads to perform the merge\n"
78 "\t-k keep around original input file on failure\n"
79 "\t-m allow input to have missing debug info\n"
80 "\t-o copy input to outfile and add CTF\n"
81 "\t-l set output container's label to specified value\n"
82 "\t-L set output container's label to value from environment\n",
83 ctfconvert_progname);
    _unchanged_portion_omitted_
247 int
248 main(int argc, char *argv[])
249 {
250     int c, ifd, err;
251     boolean_t keep = B_FALSE;
252     uint_t flags = 0;
253     uint_t nthreads = CTFCONVERT_DEFAULT_NTHREADS;
254     const char *outfile = NULL;
255     const char *label = NULL;
256     const char *infile = NULL;
257     char *tmpfile;
258     ctf_file_t *ofp;
259     long argj;
260     char *eptr;
261     char buf[4096];
262     boolean_t optx = B_FALSE;
263     boolean_t ignore_non_c = B_FALSE;
265     ctfconvert_progname = basename(argv[0]);
267     ctfconvert_altexec(argv);
269     while ((c = getopt(argc, argv, ":ij:kl:L:mo:X")) != -1) {
270     while ((c = getopt(argc, argv, ":j:kl:L:o:iX")) != -1) {
271         switch (c) {
272             case 'i':
273                 ignore_non_c = B_TRUE;
274             case 'k':
275                 keep = B_TRUE;
276                 break;
277             case 'l':
278                 label = optarg;
279         }
280     }
```



```
390     }
391     ctf_close(ofp);

393     if (tmpfile != NULL) {
394         if (rename(tmpfile, infile) != 0) {
395             int e = errno;
396             (void) unlink(outfile);
397             if (keep == B_FALSE)
398                 (void) unlink(infile);
399             ctfconvert_fatal("failed to rename temporary file: "
400                             "%s\n", strerror(e));
401         }
402     }
403     free(tmpfile);

405     return (CTFCONVERT_OK);
406 }
unchanged_portion_omitted
```

new/usr/src/cmd/ctfmerge/ctfmerge.c

1

```
*****
11748 Thu Apr 25 02:20:39 2019
new/usr/src/cmd/ctfmerge/ctfmerge.c
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 at
9  * http://www.illumos.org/license/CDDL.
10 */

12 /*
13  * Copyright (c) 2019, Joyent, Inc.
14  * Copyright (c) 2017, Joyent, Inc.
15 */

16 /*
17  * merge CTF containers
18 */

20 #include <stdio.h>
21 #include <libctf.h>
22 #include <sys/stat.h>
23 #include <sys/types.h>
24 #include <fcntl.h>
25 #include <errno.h>
26 #include <strings.h>
27 #include <assert.h>
28 #include <unistd.h>
29 #include <sys/fcntl.h>
30 #include <stdlib.h>
31 #include <libelf.h>
32 #include <gelf.h>
33 #include <sys/mman.h>
34 #include <libgen.h>
35 #include <stdarg.h>
36 #include <limits.h>

38 static char *g_progname;
39 static char *g_unique;
40 static char *g_outfile;
41 static boolean_t g_req;
42 static uint_t g_nctf;

43 #define CTFMERGE_OK      0
44 #define CTFMERGE_FATAL  1
45 #define CTFMERGE_USAGE  2

47 #define CTFMERGE_DEFAULT_NTHREADS      8
48 #define CTFMERGE_ALTEXEC              "CTFMERGE_ALTEXEC"

50 static void
51 ctfmerge_fatal(const char *fmt, ...)
```

new/usr/src/cmd/ctfmerge/ctfmerge.c

2

```
52 {
53     va_list ap;

55     (void) fprintf(stderr, "%s: ", g_progname);
56     va_start(ap, fmt);
57     (void) vfprintf(stderr, fmt, ap);
58     va_end(ap);

60     if (g_outfile != NULL)
61         (void) unlink(g_outfile);

63     exit(CTFMERGE_FATAL);
64 }

66 /*
67  * We failed to find CTF for this file, check if it's OK. If we're not derived
68  * from C, or we have the -m option, we let missing CTF pass.
69  */
70 static void
71 ctfmerge_check_for_c(const char *name, Elf *elf, uint_t flags)
72 static boolean_t
73 ctfmerge_expect_ctf(const char *name, Elf *elf)
74 {
75     char errmsg[1024];
76     Elf_Scn *scn, *strscn;
77     Elf_Data *data, *strdata;
78     GElf_Shdr shdr;
79     ulong_t i;

81     if (flags & CTF_ALLOW_MISSING_DEBUG)
82         return;
83     if (g_req == B_FALSE)
84         return (B_FALSE);

86     switch (ctf_has_c_source(elf, errmsg, sizeof (errmsg))) {
87     case CHR_ERROR:
88         ctfmerge_fatal("failed to open %s: %s\n", name, errmsg);
89         scn = NULL;
90         while ((scn = elf_nextscn(elf, scn)) != NULL) {
91             if (gelf_getshdr(scn, &shdr) == NULL) {
92                 ctfmerge_fatal("failed to get section header for file "
93                     "%s: %s\n", name, elf_errmsg(elf_errno()));
94             }

96             if (shdr.sh_type == SHT_SYMTAB)
97                 break;
98         }

100     case CHR_NO_C_SOURCE:
101         return;
102     if (scn == NULL)
103         return (B_FALSE);

105     default:
106         ctfmerge_fatal("failed to open %s: %s\n", name,
107             ctf_errmsg(ECTF_NOCTFDATA));
108         break;
109     if ((strscn = elf_getscn(elf, shdr.sh_link)) == NULL)
110         ctfmerge_fatal("failed to get section header for file %s: %s\n",
111             name, elf_errmsg(elf_errno()));

113     if ((data = elf_getdata(scn, NULL)) == NULL)
114         ctfmerge_fatal("failed to read symbol table for %s: %s\n",
115             name, elf_errmsg(elf_errno()));

117     if ((strdata = elf_getdata(strscn, NULL)) == NULL)
```

```

101     ctfmerge_fatal("failed to read string table for %s: %s\n",
102                   name, elf_errmsg(elf_errno()));
103
104     for (i = 0; i < shdr.sh_size / shdr.sh_entsize; i++) {
105         GElf_Sym sym;
106         const char *file;
107         size_t len;
108
109         if (gelf_getsym(data, i, &sym) == NULL)
110             ctfmerge_fatal("failed to read symbol table entry %lu "
111                           "for %s: %s\n", i, name, elf_errmsg(elf_errno()));
112
113         if (GELF_ST_TYPE(sym.st_info) != STT_FILE)
114             continue;
115
116         file = (const char *)((uintptr_t)strdata->d_buf + sym.st_name);
117         len = strlen(file);
118         if (len < 2 || name[len - 2] != '.')
119             continue;
120
121         if (name[len - 1] == 'c')
122             return (B_TRUE);
123     }
124
125     return (B_FALSE);
126 }
127
128 /*
129  * Go through and construct enough information for this Elf Object to try and do
130  * a ctf_bufopen().
131  */
132 static int
133 ctfmerge_elfopen(const char *name, Elf *elf, ctf_merge_t *cmh, uint_t flags)
134 static void
135 ctfmerge_elfopen(const char *name, Elf *elf, ctf_merge_t *cmh)
136 {
137     GElf_Ehdr ehdr;
138     GElf_Shdr shdr;
139     Elf_Scn *scn;
140     Elf_Data *ctf_data, *str_data, *sym_data;
141     ctf_sect_t ctfsect, symsect, strsect;
142     ctf_file_t *fp;
143     int err;
144
145     if (gelf_getehdr(elf, &ehdr) == NULL)
146         ctfmerge_fatal("failed to get ELF header for %s: %s\n",
147                       name, elf_errmsg(elf_errno()));
148
149     bzero(&ctfsect, sizeof (ctf_sect_t));
150     bzero(&symsect, sizeof (ctf_sect_t));
151     bzero(&strsect, sizeof (ctf_sect_t));
152
153     scn = NULL;
154     while ((scn = elf_nextscn(elf, scn)) != NULL) {
155         const char *sname;
156
157         if (gelf_getshdr(scn, &shdr) == NULL)
158             ctfmerge_fatal("failed to get section header for "
159                           "file %s: %s\n", name, elf_errmsg(elf_errno()));
160
161         sname = elf_strptr(elf, ehdr.e_shstrndx, shdr.sh_name);
162         if (shdr.sh_type == SHT_PROGBITS &&
163             strcmp(sname, ".SUNW_ctf") == 0) {
164             ctfsect.cts_name = sname;
165             ctfsect.cts_type = shdr.sh_type;
166             ctfsect.cts_flags = shdr.sh_flags;
167         }
168     }
169 }

```

```

130     ctfsect.cts_size = shdr.sh_size;
131     ctfsect.cts_entsize = shdr.sh_entsize;
132     ctfsect.cts_offset = (off64_t)shdr.sh_offset;
133
134     ctf_data = elf_getdata(scn, NULL);
135     if (ctf_data == NULL)
136         ctfmerge_fatal("failed to get ELF CTF "
137                       "data section for %s: %s\n", name,
138                       elf_errmsg(elf_errno()));
139     ctfsect.cts_data = ctf_data->d_buf;
140 } else if (shdr.sh_type == SHT_SYMTAB) {
141     Elf_Scn *strscn;
142     GElf_Shdr strhdr;
143
144     symsect.cts_name = sname;
145     symsect.cts_type = shdr.sh_type;
146     symsect.cts_flags = shdr.sh_flags;
147     symsect.cts_size = shdr.sh_size;
148     symsect.cts_entsize = shdr.sh_entsize;
149     symsect.cts_offset = (off64_t)shdr.sh_offset;
150
151     if ((strscn = elf_getscn(elf, shdr.sh_link)) == NULL ||
152         gelf_getshdr(strscn, &strhdr) == NULL)
153         ctfmerge_fatal("failed to get "
154                       "string table for file %s: %s\n", name,
155                       elf_errmsg(elf_errno()));
156
157     strsect.cts_name = elf_strptr(elf, ehdr.e_shstrndx,
158                                 strhdr.sh_name);
159     strsect.cts_type = strhdr.sh_type;
160     strsect.cts_flags = strhdr.sh_flags;
161     strsect.cts_size = strhdr.sh_size;
162     strsect.cts_entsize = strhdr.sh_entsize;
163     strsect.cts_offset = (off64_t)strhdr.sh_offset;
164
165     sym_data = elf_getdata(scn, NULL);
166     if (sym_data == NULL)
167         ctfmerge_fatal("failed to get ELF CTF "
168                       "data section for %s: %s\n", name,
169                       elf_errmsg(elf_errno()));
170     symsect.cts_data = sym_data->d_buf;
171
172     str_data = elf_getdata(strscn, NULL);
173     if (str_data == NULL)
174         ctfmerge_fatal("failed to get ELF CTF "
175                       "data section for %s: %s\n", name,
176                       elf_errmsg(elf_errno()));
177     strsect.cts_data = str_data->d_buf;
178 }
179
180 if (ctfsect.cts_type == SHT_NULL) {
181     ctfmerge_check_for_c(name, elf, flags);
182     return (ENOENT);
183     if (ctfmerge_expect_ctf(name, elf) == B_FALSE)
184         return;
185     ctfmerge_fatal("failed to open %s: %s\n", name,
186                   ctf_errmsg(ECTF_NOCTFDATA));
187 }
188
189 if (symsect.cts_type != SHT_NULL && strsect.cts_type != SHT_NULL) {
190     fp = ctf_bufopen(&ctfsect, &symsect, &strsect, &err);
191 } else {
192     fp = ctf_bufopen(&ctfsect, NULL, NULL, &err);
193 }

```

```

192     if (fp == NULL) {
230         if (ctfmerge_expect_ctf(name, elf) == B_TRUE) {
193             ctfmerge_fatal("failed to open file %s: %s\n",
194                 name, ctf_errmsg(err));
195         }
234     } else {
197         if ((err = ctf_merge_add(cmh, fp)) != 0) {
198             ctfmerge_fatal("failed to add input %s: %s\n",
199                 name, ctf_errmsg(err));
200         }
202         g_nctf++;
203         return (0);
204     }
206 static void
207 ctfmerge_read_archive(const char *name, int fd, Elf *elf,
208     ctf_merge_t *cmh, uint_t flags)
245     ctf_merge_t *cmh)
209 {
247     Elf *aelf;
210     Elf_Cmd cmd = ELF_C_READ;
211     int cursec = 1;
212     Elf *aelf;
250     char *nname;
214     while ((aelf = elf_begin(fd, cmd, elf)) != NULL) {
215         char *nname = NULL;
216         Elf_Arhdr *arhdr;
254         boolean_t leakelf = B_FALSE;
218         if ((arhdr = elf_getarhdr(aelf)) == NULL)
219             ctfmerge_fatal("failed to get archive header %d for "
220                 "%s: %s\n", cursec, name, elf_errmsg(elf_errno()));
222         cmd = elf_next(aelf);
224         if *(arhdr->ar_name) == '/')
225             goto next;
227         if (asprintf(&nname, "%s.%s.%d", name, arhdr->ar_name,
228             cursec) < 0)
229             ctfmerge_fatal("failed to allocate memory for archive "
230                 "%d of file %s\n", cursec, name);
232         switch (elf_kind(aelf)) {
233         case ELF_K_AR:
234             ctfmerge_read_archive(nname, fd, aelf, cmh, flags);
270             ctfmerge_read_archive(nname, fd, aelf, cmh);
271             free(nname);
235             break;
236         case ELF_K_ELF:
237             /* ctfmerge_elfopen() takes ownership of aelf. */
238             if (ctfmerge_elfopen(nname, aelf, cmh, flags) == 0)
239                 aelf = NULL;
274             ctfmerge_elfopen(nname, aelf, cmh);
275             free(nname);
276             leakelf = B_TRUE;
240             break;
241         default:
242             ctfmerge_fatal("unknown elf kind (%d) in archive %d "
243                 "for %s\n", elf_kind(aelf), cursec, name);
244             break;
245     }

```

```

247 next:
284         cmd = elf_next(aelf);
285         if (leakelf == B_FALSE)
248             (void) elf_end(aelf);
249         free(nname);
250         cursec++;
251     }
252 }
254 static void
255 ctfmerge_file_add(ctf_merge_t *cmh, const char *file, uint_t flags)
256 {
257     Elf *e;
258     int fd;
260     if ((fd = open(file, O_RDONLY)) < 0) {
261         ctfmerge_fatal("failed to open file %s: %s\n",
262             file, strerror(errno));
263     }
265     if ((e = elf_begin(fd, ELF_C_READ, NULL)) == NULL) {
266         (void) close(fd);
267         ctfmerge_fatal("failed to open %s: %s\n",
268             file, elf_errmsg(elf_errno()));
269     }
271     switch (elf_kind(e)) {
272     case ELF_K_AR:
273         ctfmerge_read_archive(file, fd, e, cmh, flags);
274         break;
276     case ELF_K_ELF:
277         /* ctfmerge_elfopen() takes ownership of e. */
278         if (ctfmerge_elfopen(file, e, cmh, flags) == 0)
279             e = NULL;
280         break;
282     default:
283         ctfmerge_fatal("unknown elf kind (%d) for %s\n",
284             elf_kind(e), file);
285     }
287     (void) elf_end(e);
288     (void) close(fd);
289 }
291 static void
292 ctfmerge_usage(const char *fmt, ...)
293 {
294     if (fmt != NULL) {
295         va_list ap;
297         (void) fprintf(stderr, "%s: ", g_progname);
298         va_start(ap, fmt);
299         (void) vfprintf(stderr, fmt, ap);
300         va_end(ap);
301     }
303     (void) fprintf(stderr, "Usage: %s [-m] [-d uniqfile] [-l label] "
304         "(void) fprintf(stderr, \"Usage: %s [-t] [-d uniqfile] [-l label] \"
305         \"[-L labelenv] [-j nthrs] -o outfile file ...\\n\"
306         \"\\n\"
307         \"\\t-d  uniquify merged output against uniqfile\\n\"
308         \"\\t-j  use nthrs threads to perform the merge\\n\"
309         \"\\t-l  set output container's label to specified value\\n\"

```



```

309     "\t-L set output container's label to value from environment\n"
310     "\t-m allow C-based input files to not have CTF\n"
311     "\t-o file to add CTF data to\n",
310     "\t-o file to add CTF data to\n"
311     "\t-t require CTF data from all inputs built from C sources\n",
312     g_progname);
313 }
    unchanged_portion_omitted

337 int
338 main(int argc, char *argv[])
339 {
340     int err, i, c, ofd;
341     uint_t nthreads = CTFMERGE_DEFAULT_NTHREADS;
342     char *tmpfile = NULL, *label = NULL;
343     int wflags = CTF_ELFWRITE_F_COMPRESS;
344     uint_t flags = 0;
345     ctf_merge_t *cmh;
346     ctf_file_t *ofp;
345     ctf_merge_t *cmh;
347     long argj;
348     char *eptr;

350     g_progname = basename(argv[0]);

352     ctfmerge_altexec(argv);

354     /*
355     * We support a subset of the old CTF merge flags, mostly for
356     * compatibility.
357     * compatibility.
358     */
358     while ((c = getopt(argc, argv, ":d:fgj:l:L:mo:t")) != -1) {
357     while ((c = getopt(argc, argv, ":d:fgj:l:L:mo:t")) != -1) {
359         switch (c) {
360             case 'd':
361                 g_unique = optarg;
362                 break;
363             case 'f':
364                 /* Silently ignored for compatibility */
365                 break;
366             case 'g':
367                 /* Silently ignored for compatibility */
368                 break;
369             case 'j':
370                 errno = 0;
371                 argj = strtol(optarg, &eptr, 10);
372                 if (errno != 0 || argj == LONG_MAX ||
373                     argj > 1024 || *eptr != '\0') {
374                     ctfmerge_fatal("invalid argument for -j: %s\n",
375                                 optarg);
376                 }
377                 nthreads = (uint_t)argj;
378                 break;
379             case 'l':
380                 label = optarg;
381                 break;
382             case 'L':
383                 label = getenv(optarg);
384                 break;
385             case 'm':
386                 flags |= CTF_ALLOW_MISSING_DEBUG;
387                 break;
388             case 'o':
389                 g_outfile = optarg;
390                 break;

```

```

391         case 't':
392             /* Silently ignored for compatibility */
393             g_req = B_TRUE;
394             break;
395         case ':':
396             ctfmerge_usage("Option -%c requires an operand\n",
397                             optopt);
398             return (CTFMERGE_USAGE);
399         case '?':
400             ctfmerge_usage("Unknown option: -%c\n", optopt);
401             return (CTFMERGE_USAGE);
402     }

404     if (g_outfile == NULL) {
405         ctfmerge_usage("missing required -o output file\n");
406         return (CTFMERGE_USAGE);
407     }

409     (void) elf_version(EV_CURRENT);

411     /*
412     * Obviously this isn't atomic, but at least gives us a good starting
413     * point.
414     */
415     if ((ofd = open(g_outfile, O_RDWR)) < 0)
416         ctfmerge_fatal("cannot open output file %s: %s\n", g_outfile,
417                       strerror(errno));

419     argc -= optind;
420     argv += optind;

422     if (argc < 1) {
423         ctfmerge_usage("no input files specified");
424         return (CTFMERGE_USAGE);
425     }

427     cmh = ctf_merge_init(ofd, &err);
428     if (cmh == NULL)
429         ctfmerge_fatal("failed to create merge handle: %s\n",
430                       ctf_errmsg(err));

432     if ((err = ctf_merge_set_nthreads(cmh, nthreads)) != 0)
433         ctfmerge_fatal("failed to set parallelism to %u: %s\n",
434                       nthreads, ctf_errmsg(err));

436     for (i = 0; i < argc; i++) {
437         ctfmerge_file_add(cmh, argv[i], flags);
438         ctf_file_t *ifp;
439         int fd;

436         if ((fd = open(argv[i], O_RDONLY)) < 0)
437             ctfmerge_fatal("failed to open file %s: %s\n",
438                             argv[i], strerror(errno));
439         ifp = ctf_fdopen(fd, &err);
440         if (ifp == NULL) {
441             Elf *e;

443             if ((e = elf_begin(fd, ELF_C_READ, NULL)) == NULL) {
444                 (void) close(fd);
445                 ctfmerge_fatal("failed to open %s: %s\n",
446                                 argv[i], ctf_errmsg(err));
447             }
448         }

449         /*
450         * It's an ELF file, check if we have an archive or if

```

```

451     * we're expecting CTF here.
452     */
453     switch (elf_kind(e)) {
454     case ELF_K_AR:
455         break;
456     case ELF_K_ELF:
457         if (ctfmerge_expect_ctf(argv[i], e) == B_TRUE) {
458             (void) elf_end(e);
459             (void) close(fd);
460             ctfmerge_fatal("failed to "
461                "open %s: file was built from C "
462                "sources, but missing CTF\n",
463                argv[i]);
464         }
465         (void) elf_end(e);
466         (void) close(fd);
467         continue;
468     default:
469         (void) elf_end(e);
470         (void) close(fd);
471         ctfmerge_fatal("failed to open %s: "
472            "unsupported ELF file type", argv[i]);
473     }
474
475     ctfmerge_read_archive(argv[i], fd, e, cmh);
476     (void) elf_end(e);
477     (void) close(fd);
478     continue;
479 }
480 (void) close(fd);
481 if ((err = ctf_merge_add(cmh, ifp)) != 0)
482     ctfmerge_fatal("failed to add input %s: %s\n",
483        argv[i], ctf_errmsg(err));
484 g_nctf++;
485 }
486
487 if (g_nctf == 0) {
488     ctf_merge_fini(cmh);
489     return (0);
490 }
491
492 if (g_unique != NULL) {
493     ctf_file_t *ufp;
494     char *base;
495
496     ufp = ctf_open(g_unique, &err);
497     if (ufp == NULL) {
498         ctfmerge_fatal("failed to open uniquify file %s: %s\n",
499            g_unique, ctf_errmsg(err));
500     }
501
502     base = basename(g_unique);
503     (void) ctf_merge_uniquify(cmh, ufp, base);
504 }
505
506 if (label != NULL) {
507     if ((err = ctf_merge_label(cmh, label)) != 0)
508         ctfmerge_fatal("failed to add label %s: %s\n", label,
509            ctf_errmsg(err));
510 }
511
512 err = ctf_merge_merge(cmh, &ofp);
513 if (err != 0)
514     ctfmerge_fatal("failed to merge types: %s\n", ctf_errmsg(err));
515 ctf_merge_fini(cmh);

```

```

470     if (asprintf(&tmpfile, "%s.ctf", g_outfile) == -1)
471         ctfmerge_fatal("ran out of memory for temporary file name\n");
472     err = ctf_elfwrite(ofp, g_outfile, tmpfile, wflags);
473     if (err == CTF_ERR) {
474         (void) unlink(tmpfile);
475         free(tmpfile);
476         ctfmerge_fatal("encountered a libctf error: %s!\n",
477            ctf_errmsg(ctf_errno(ofp)));
478     }
479
480     if (rename(tmpfile, g_outfile) != 0) {
481         (void) unlink(tmpfile);
482         free(tmpfile);
483         ctfmerge_fatal("failed to rename temporary file: %s\n",
484            strerror(errno));
485     }
486     free(tmpfile);
487
488     return (CTFMERGE_OK);
489 }

```

unchanged_portion_omitted

new/usr/src/cmd/mdb/Makefile.module

1

```
*****
8770 Thu Apr 25 02:20:39 2019
new/usr/src/cmd/mdb/Makefile.module
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 # Copyright (c) 2013 by Delphix. All rights reserved.
26 # Copyright (c) 2018, Joyent, Inc.
27 # Copyright (c) 2019 Carlos Neira <cneirabustos@gmail.com>
28 # Copyright 2019 OmniOS Community Edition (OmniOSce) Association.
29 # Copyright (c) 2019, Joyent, Inc.
30 #
31 .KEEP_STATE:
32 .SUFFIXES:
33 #
34 include $(SRC)/cmd/mdb/Makefile.tools
35 #
36 #
37 # Make sure we're getting a consistent execution environment for the
38 # embedded scripts.
39 #
40 SHELL= /usr/bin/ksh93
41 #
42 $(KMOD_SOURCES_DIFFERENT)KMODSRCS = $(MODSRCS)
43 $(KMOD_SOURCES_DIFFERENT)KMODASMSRCS = $(MODASMSRCS)
44 #
45 MODOBJS = $(MODSRCS:%.c=dmod/%.o) $(MODASMSRCS:%.s=dmod/%.o)
46 KMODOBJS = $(KMODSRCS:%.c=kmod/%.o) $(KMODASMSRCS:%.s=kmod/%.o)
47 #
48 MODNAME_cmd = if [ -n "$(MODULE_NAME)" ]; then print $(MODULE_NAME); else print
49 MODNAME = $(MODNAME_cmd:sh)
50 KMODULE = $(MODNAME)
51 #
52 MODFILE = dmod/$(MODULE)
```

new/usr/src/cmd/mdb/Makefile.module

2

```
53 KMODFILE = kmod/$(KMODULE)
54 #
55 #
56 # The mess below is designed to pick the right set of objects to build and/or
57 # lint. We have three flavors:
58 #
59 # 1. proc and raw modules. Only $(MODOBJS) are built.
60 # 2. kvm modules for systems without kmdb. Only $(MODOBJS) are built.
61 # 3. kvm modules for systems with kmdb. $(MODOBJS) and $(KMODOBJS) are built.
62 #
63 # Complicating matters, we'd like to make the distinction between 2 and 3 before
64 # this Makefile is loaded. By default, we'll assume that all kvm modules should
65 # be built for kmdb. If, however, the user sets $(MODULE_BUILD_TYPE) to 'mdb',
66 # the kmdb variant of the module won't be built.
67 #
68 #
69 # Which flavors are to be built?
70 TARGETS_kvm_type_ = both # Build both if $(MODULE_BUILD_TYPE) is unset
71 TARGETS_kvm_type_kmdb = both
72 TARGETS_kvm_type_mdb = mdb
73 TARGETS_kvm_type = $(TARGETS_kvm_type_$(MODULE_BUILD_TYPE))
74 #
75 # What should we build?
76 TARGETS_kvm_kmdb = $(KMODFILE)
77 TARGETS_kvm_mdb = $(MODFILE)
78 TARGETS_kvm_both = $(TARGETS_kvm_mdb) $(TARGETS_kvm_kmdb)
79 TARGETS_kvm = $(TARGETS_kvm_$(TARGETS_kvm_type))
80 TARGETS_proc = $(MODFILE)
81 TARGETS_raw = $(MODFILE)
82 TARGETS = $(TARGETS_$(MDBTGT))
83 #
84 # Where should we install that which we've built?
85 ROOTTGTS_kvm_type = $(TARGETS_kvm_type) # overridden by mdb_ks
86 ROOTTGTS_kvm_kmdb = $(ROOTKMOD)/$(KMODULE)
87 ROOTTGTS_kvm_mdb = $(ROOTMOD)/$(MODULE)
88 ROOTTGTS_kvm_both = $(ROOTTGTS_kvm_mdb) $(ROOTTGTS_kvm_kmdb)
89 ROOTTGTS_kvm = $(ROOTTGTS_kvm_$(ROOTTGTS_kvm_type))
90 ROOTTGTS_proc = $(ROOTMOD)/$(MODULE)
91 ROOTTGTS_raw = $(ROOTMOD)/$(MODULE)
92 ROOTTGTS = $(ROOTTGTS_$(MDBTGT))
93 #
94 # What should we lint?
95 KLINTOBJS = $(KMODOBJS:%.o=%.ln)
96 LINTOBJS = $(MODOBJS:%.o=%.ln)
97 #
98 LINTFILES_kvm_type = $(TARGETS_kvm_type)
99 LINTFILES_kvm_both = $(KLINTOBJS) $(LINTOBJS)
100 LINTFILES_kvm_mdb = $(LINTOBJS)
101 LINTFILES_kvm = $(LINTFILES_kvm_$(LINTFILES_kvm_type))
102 LINTFILES_proc = $(LINTOBJS)
103 LINTFILES_raw = $(LINTOBJS)
104 LINTFILES = $(LINTFILES_$(MDBTGT))
105 #
106 #
107 # Python specific flags. To try and make life easier for folks how are
108 # building with an LFS python, we attempt to use -isystem when it's
109 # available.
110 #
111 PYCPPFLAGS = -gcc=-isystem -gcc=$(ADJUNCT_PROTO)/usr/include/pyth
112 PYCPPFLAGS += -cc=-I$(ADJUNCT_PROTO)/usr/include/python$(PYTHON_VE
113 PYLNLFLAGS = -I$(ADJUNCT_PROTO)/usr/include/python$(PYTHON_VERSION)
114 #
115 kvm_TGTFLAGS = -D_KERNEL
116 proc_TGTFLAGS = -D_USER
117 #
118 CSTD = $(CSTD_GNU99)
```

```

120 CFLAGS           += $(CCVERBOSE)
121 CFLAGS64          += $(CCVERBOSE)
122 CPPFLAGS          += $( $(MDBTGT)_TGTFLAGS) -I../../common
123 LDFLAGS           += $(ZTEXT)
124 LDFLAGS64         += $(ZTEXT)
125 ASFLAGS           += -P
126 AS_CPPFLAGS       += -D_ASM

128 SMOFF += all_func_returns,index_overflow

130 # Module type-specific compiler flags
131 $(MODOBJS) := CFLAGS += $(C_BIGPICFLAGS) $(XREGSFLAG)
132 $(MODOBJS) := CFLAGS64 += $(C_BIGPICFLAGS) $(XREGSFLAG)
133 $(KMODOBJS) $(KLINTOBJS) := CPPFLAGS += -D_KMDB
134 $(KMODOBJS) := V9CODESIZE = $(CCABS32)
135 $(KMODOBJS) := DTS_ERRNO =

137 # Modules aren't allowed to export symbols
138 MAPFILE          = $(SRC)/cmd/mdb/common/modules/conf/mapfile

140 # Modules typically make external references. To provide for -zdefs use
141 # and clean ldd(1) processing, explicitly define all external references.
142 MAPFILE-EXT      = $(SRC)/cmd/mdb/common/modules/conf/mapfile-extern

144 #
145 # kmdb is a kernel module, so we'll use the kernel's build flags.
146 $(KMODOBJS) := CFLAGS64 += $(STAND_FLAGS_64)

148 #
149 # Override this to pull source files from another directory
150 #
151 MODSRCS_DIR = ../../../../common/modules/genunix

153 all: $$ (TARGETS)

155 install: all $$ (ROOTTGTS)

157 dmods: install

159 clean.lint:
160     $(RM) $(LINTFILES) $(MODSRCS:.c=.ln)

162 clean:
163     $(RM) $(MODOBJS) $(KMODOBJS) $(CLEANFILES)

165 clobber: clean clean.lint
166     $(RM) $(MODFILE) $(KMODFILE) $(CLOBBERFILES)

168 lint: $$ (LINTFILES)

170 .NO_PARALLEL:
171 .PARALLEL: $(MODOBJS) $(KMODOBJS) mdb_tgt kmdb_tgt dmod kmod \
172     $(TARGETS) $(LINTFILES)

174 $(MODFILE): dmod .WAIT $(MODOBJS) $$ (MAPFILE-EXT)
175     $(LINK.c) $(ZDEFS) $(ZIGNORE) $(MAPFILE-EXT:%=-M%) $(GSHARED) \
176     $(MODOBJS) -o $$@ $(LDLIBS) -lc -lproc
177     $(CTFMERGE) -L VERSION -o $$@ $(MODOBJS)
178     $(POST_PROCESS_SO)

180 #
181 # kmdb dmods must *not* stray from the module API. To ensure that they don't,
182 # we try to link them, at build time, against an object that exports the symbols
183 # that they can legally use. The link test object is, however, only built when
184 # kmdb itself is built. Requiring module developers to build kmdb first would

```

```

185 # be painful, so by default, module-level builds don't do the link test (the
186 # $(POUND_SIGN) assignment below takes care of that). Builds of the entire
187 # tree can, however, guarantee the construction of kmdb first, and as such can
188 # override the setting of $(KMDB_LINKTEST_ENABLE). This override causes the
189 # link test to be run.
190 #
191 # Developers wanting to force a link test for a single module can use the
192 # 'linktest' target from within a module directory.
193 #
194 LINKTESTOBJ = $(KMDBDIR)/kmdb_modlinktest.o

196 KMDB_LINKTEST = \
197     $(LD) $(ZDEFS) -dy -r -o $$@.linktest $(KMODOBJS) \
198     $(STANDOBJS) $(LINKTESTOBJ) && \
199     $(RM) $$@.linktest

201 KMDB_LINKTEST_ENABLE=$(POUND_SIGN)
202 $(KMDB_LINKTEST_ENABLE)KMDB_LINKTEST_CMD = $(KMDB_LINKTEST)

204 #
205 # Ensure that dmods don't use floating point
206 #
207 KMDB_FPTEST_CMD = $(KMDB_FPTEST)

209 $(KMODFILE): kmod .WAIT $(KMODOBJS) $(MAPFILE)
210     $(LD) -dy -r $(MAPFILE:%=-M%) -Nmisc/kmdbmod -o $$@ $(KMODOBJS) \
211     $(STANDOBJS)
212     $(KMDB_LINKTEST_CMD)
213     $(KMDB_FPTEST_CMD)
214     $(CTFMERGE) -f -L VERSION -o $$@ $(KMODOBJS)
215     $(POST_PROCESS)
216     $(SETDYNFLAG) -f DF_1_NOKSYMS $$@

218 linktest: linktest_check .WAIT kmod .WAIT $(KMODOBJS)
219     $(KMDB_LINKTEST)

221 linktest_check:
222     @if [ "$(MDBTGT)" != "kvm" ] ; then \
223         echo "ERROR: linktest is not supported non-kvm/disasm dmods" \
224             >&2 ; \
225         exit 1 ; \
226     fi

228 #
229 # Dynamic rules for object construction
230 #
231 dmod/%.o kmod/%.o: %.c
232     $(COMPILE.c) -o $$@ $<
233     $(CTFCONVERT_O)

235 dmod/%.o kmod%.o: %.s
236     $(COMPILE.s) -o $$@ $<
237     $(CTFCONVERT_O)

238 dmod/%.o kmod/%.o: ../%.c
239     $(COMPILE.c) -o $$@ $<
240     $(CTFCONVERT_O)

242 dmod/%.o kmod%.o: ../%.s
243     $(COMPILE.s) -o $$@ $<
244     $(CTFCONVERT_O)

245 dmod/%.o kmod/%.o: ../../../../common/modules/$(MODNAME)/%.c
246     $(COMPILE.c) -o $$@ $<
247     $(CTFCONVERT_O)

```

```
249 dmod/%.o kmod%.o: ../../../../common/modules/$(MODNAME)/%.s
250 $(COMPILE.s) -o $@ $<
253 $(CTFCONVERT_O)

252 dmod/%.o kmod/%.o: $$ (MODSRCS_DIR)/%.c
253 $(COMPILE.c) -o $@ $<
254 $(CTFCONVERT_O)

256 dmod/%.o kmod%.o: $$ (MODSRCS_DIR)/%.s
257 $(COMPILE.s) -o $@ $<
261 $(CTFCONVERT_O)

259 #
260 # Lint
261 #
262 dmod/%.ln kmod/%.ln: %.c
263 $(LINT.c) -dirout=$(@D) -c $<

265 dmod/%.ln kmod/%.ln: %.s
266 $(LINT.s) -dirout=$(@D) -c $<

268 dmod/%.ln kmod/%.ln: ../%.c
269 $(LINT.c) -dirout=$(@D) -c $<

271 dmod/%.ln kmod/%.ln: ../%.s
272 $(LINT.s) -dirout=$(@D) -c $<

274 dmod/%.ln kmod/%.ln: ../../../../common/modules/$(MODNAME)/%.c
275 $(LINT.c) -dirout=$(@D) -c $<

277 dmod/%.ln kmod/%.ln: ../../../../common/modules/$(MODNAME)/%.s
278 $(LINT.s) -dirout=$(@D) -c $<

280 dmod/%.ln kmod/%.ln: $$ (MODSRCS_DIR)/%.c
281 $(LINT.c) -dirout=$(@D) -c $<

283 dmod/%.ln kmod/%.ln: $$ (MODSRCS_DIR)/%.s
284 $(LINT.s) -dirout=$(@D) -c $<

286 #
287 # Installation targets
288 #

290 $(ROOT)/usr/lib/mdb/$(MDBTGT): $(ROOT)/usr/lib/mdb
291 $(INS.dir)

293 $(ROOT)/usr/lib/mdb:
294 $(INS.dir)

296 $(ROOT)/kernel/kmdb:
297 $(INS.dir)

299 $(ROOTMOD)/$(MODULE): $(ROOTMOD)

301 $(ROOTKMOD)/$(KMODULE): $(ROOTKMOD)

303 kmod dmod:
304 -@mkdir -p $@
```

new/usr/src/cmd/mdb/common/modules/genunix/Makefile.files

1

1827 Thu Apr 25 02:20:40 2019

new/usr/src/cmd/mdb/common/modules/genunix/Makefile.files

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 2011 Nexenta Systems, Inc. All rights reserved.
23 # Copyright (c) 1999, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright (c) 2019, Joyent, Inc.
24 # Copyright 2016 Joyent, Inc.
25 # Copyright (c) 2013 by Delphix. All rights reserved.
26 #
```

```
28 #
29 # This file simply contains the list of sources files compiled together
30 # to create the genunix mdb module. Having them in one place saves
31 # a bunch of unnecessary replication.
32 #
```

```
33 GENUNIX_SRCS = \
34     avl.c \
35     bio.c \
36     bitset.c \
37     combined.c \
38     contract.c \
39     cpupart.c \
40     cred.c \
41     ctxop.c \
42     cyclic.c \
43     damap.c \
44     ddi_periodic.c \
45     devinfo.c \
46     dist.c \
47     dnlc.c \
48     findstack.c \
49     findstack_subr.c \
50     fm.c \
51     gcore.c \
51     gcore_isadep.c \
```

new/usr/src/cmd/mdb/common/modules/genunix/Makefile.files

2

```
52     genunix.c \
53     group.c \
54     hotplug.c \
55     irm.c \
56     kgrep.c \
57     kmem.c \
58     ldi.c \
59     leaky.c \
60     leaky_subr.c \
61     lgrp.c \
62     list.c \
63     log.c \
64     mdi.c \
65     memory.c \
66     mmd.c \
67     modhash.c \
68     ndievents.c \
69     net.c \
70     netstack.c \
71     nvpair.c \
72     pg.c \
73     rctl.c \
74     sobj.c \
75     streams.c \
76     sysevent.c \
77     taskq.c \
78     thread.c \
79     tsd.c \
80     tsol.c \
81     vfs.c \
82     zone.c \
```

new/usr/src/cmd/mdb/intel/amd64/genunix/Makefile

1

2330 Thu Apr 25 02:20:40 2019
new/usr/src/cmd/mdb/intel/amd64/genunix/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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

22 #
23 #
24 # Copyright 2007 Sun Microsystems, Inc. All rights reserved.
25 # Use is subject to license terms.
26 #
27 # Copyright (c) 2013 by Delphix. All rights reserved.
28 #
29 # Copyright (c) 2019, Joyent, Inc.
30 #

32 MODULE = genunix.so
33 MDBTGT = kvm

35 include ../../../../common/modules/genunix/Makefile.files

37 COMMONSRCS = \
38     $(GENUNIX_SRCS)

40 KMODSRCS = \
41     $(COMMONSRCS)

43 MODSRCS = \
44     $(COMMONSRCS) \
45     gcore.c \
46     typegraph.c

48 #
49 # This signals that $(KMODSRCS) != $(MODSRCS). Typegraph is not usable under
50 # kmdb. As such, we don't bother compiling it.
51 KMOD_SOURCES_DIFFERENT=$(POUND_SIGN)

53 include ../../../../Makefile.cmd
```

new/usr/src/cmd/mdb/intel/amd64/genunix/Makefile

2

```
54 include ../../../../Makefile.cmd.64
55 include ../../../../Makefile.amd64
56 include ../../../../Makefile.module

58 dmod$(MODULE) := LDLIBS += -lm -lproc

60 #
61 # We are not actually hardwiring some dependency on i86pc, we just need to
62 # include sys/param.h with _MACHDEP defined, and this forces the inclusion of
63 # machparam.h, even though we don't use anything there. This is a temporary
64 # kludge until we invent -DDONTINCLUDEMACHPARAM or something.
65 #
66 CPPFLAGS += -I$(SRC)/uts/i86pc
67 CPPFLAGS += -I$(SRC)/uts/i86xpv

69 # Needed to include c2/audit.h (from cred.h)
70 CPPFLAGS += -I$(SRC)/uts/common

72 # Needed to find include file mutex_impl.h
73 CPPFLAGS += -I$(SRC)/uts/intel

75 CERRWARN += -_gcc=-Wno-char-subscripts
76 CERRWARN += -_gcc=-Wno-unused-label
77 CERRWARN += -_gcc=-Wno-uninitialized
78 CERRWARN += -_gcc=-Wno-parentheses
79 CERRWARN += -_gcc=-Wno-type-limits

81 LINTFLAGS64 += -erroff=E_EMPTY_TRANSLATION_UNIT

83 MODSRCS_DIR = ../../../../intel/modules/genunix
```

new/usr/src/cmd/sgs/rtld/Makefile.targ

1

2506 Thu Apr 25 02:20:40 2019
new/usr/src/cmd/sgs/rtld/Makefile.targ
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 #
23 # Copyright 2010 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 #
26 # Copyright 2019 OmniOS Community Edition (OmniOSce) Association.
27 #
28 # Copyright (c) 2019, Joyent, Inc.
29 #
30 #
31 # Don't generate separate sections for this module
32 pics/debugdata.o := XFFLAG=
33 #
34 # Since we use RTLD rather than DYNLIB, we must explicitly hook in ctfmerge.
35 $(RTLD) := CTFMERGE_POST = $(CTFMERGE_LIB)
36 #
37 pics/%.o: %.s
38 $(BUILDD.s)
39 $(POST_PROCESS_S_O)
40 $(POST_PROCESS_O)
41 #
42 pics/%.o: $(CRTSRCS)/%.s
43 $(BUILDD.s)
44 $(POST_PROCESS_S_O)
45 $(POST_PROCESS_O)
46 #
47 pics/%.o: %.c
48 $(COMPILE.c) -o $@ $<
49 $(POST_PROCESS_O)
50 #
51 pics/%.o: ../common/%.c
52 $(COMPILE.c) -o $@ $<
53 $(POST_PROCESS_O)
```

new/usr/src/cmd/sgs/rtld/Makefile.targ

2

```
53 pics/%.o: $(VAR_AVLDIR)/%.c
54 $(COMPILE.c) -o $@ $<
55 $(POST_PROCESS_O)
56 #
57 pics/%.o: $(VAR_DTRDIR)/%.c
58 $(COMPILE.c) -o $@ $<
59 $(POST_PROCESS_O)
60 #
61 pics/%.o: $(SGSTOOLS)/common/%.c
62 $(COMPILE.c) -o $@ $<
63 $(POST_PROCESS_O)
64 #
65 pics/%.o: $(SRCBASE)/uts/$(PLAT)/krtld/%.c
66 $(COMPILE.c) -o $@ $<
67 $(POST_PROCESS_O)
68 #
69 pics/elfcap.o: $(ELFCAP)/elfcap.c
70 $(COMPILE.c) -o $@ $(ELFCAP)/elfcap.c
71 $(POST_PROCESS_O)
72 #
73 $(RTLD): pics $(PICS) $(CRTS)
74 $(LD_USE_PROTO)ld -o $@ -dy -G $(DYNFLAGS) $(CRTI) $(PICS) \
75 $(LDLIBS) $(CRTN)
76 $(POST_PROCESS_SO)
77 #
78 delete: -$(RM) $(RTLD)
79 #
80 $(ROOTFS_LIBDIR)/%: %
81 $(INS.file)
82 #
83 $(ETCLIBDIR)/%: %
84 $(INS.file)
85 #
86 #
87 include $(SRC)/lib/Makefile.targ
88 include $(SRC)/cmd/sgs/Makefile.targ
89 #
90 # Derived source and header files (messaging).
91 #
92 catalog: $(BLTMESG)
93 #
94 chkmsg := SGMSGCHK=../common/rtld.chk.msg
95 #
96 chkmsg: $(SRCS) $(CHKSRCS)
97 sh $(CHKMSG) $(CHKMSGFLAGS) $(SRCS) $(CHKSRCS)
98 #
99 $(BLTDEFS) + \
100 $(BLTDATA) : $(SGMSGTARG)
101 $(SGMSG) $(SGMSGFLAGS2) $(SGMSGTARG)
102 #
103 $(BLTMESG) : $(SGMSGGALL)
104 $(SGMSG) $(SGMSGFLAGS1) $(SGMSGGALL)
```


new/usr/src/common/ctf/ctf_error.c

1

```
*****
4637 Thu Apr 25 02:20:40 2019
new/usr/src/common/ctf/ctf_error.c
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23  * Copyright 2003 Sun Microsystems, Inc. All rights reserved.
24  * Use is subject to license terms.
25 */
26 /*
27  * Copyright (c) 2019, Joyent, Inc.
28  * Copyright (c) 2015, Joyent, Inc.
29 */
30 #include <ctf_impl.h>
31
32 static const char *const_ctf_errlist[] = {
33     "File is not in CTF or ELF format", /* ECTF_FMT */
34     "File uses more recent ELF version than libctf", /* ECTF_ELFVERS */
35     "File uses more recent CTF version than libctf", /* ECTF_CTFVERS */
36     "File is a different endianness than libctf", /* ECTF_ENDIAN */
37     "Symbol table uses invalid entry size", /* ECTF_SYMTAB */
38     "Symbol table data buffer is not valid", /* ECTF_SYMBAD */
39     "String table data buffer is not valid", /* ECTF_STRBAD */
40     "File data structure corruption detected", /* ECTF_CORRUPT */
41     "File does not contain CTF data", /* ECTF_NOCTFDATA */
42     "Buffer does not contain CTF data", /* ECTF_NOCTFBUF */
43     "Symbol table information is not available", /* ECTF_NOSYMTAB */
44     "Type information is in parent and unavailable", /* ECTF_NOPARENT */
45     "Cannot import types with different data model", /* ECTF_DMODEL */
46     "Failed to mmap a needed data section", /* ECTF_MMAP */
47     "Decompression package SUNWzlib not installed", /* ECTF_ZMISSING */
48     "Failed to initialize decompression library", /* ECTF_ZINIT */
49     "Failed to allocate decompression buffer", /* ECTF_ZALLOC */
50     "Failed to decompress CTF data", /* ECTF_DECOMPRESS */
51     "External string table is not available", /* ECTF_STRTAB */
52     "String name offset is corrupt", /* ECTF_BADNAME */

```

new/usr/src/common/ctf/ctf_error.c

2

```
53     "Invalid type identifier", /* ECTF_BADID */
54     "Type is not a struct or union", /* ECTF_NOTSOU */
55     "Type is not an enum", /* ECTF_NOTENUM */
56     "Type is not a struct, union, or enum", /* ECTF_NOTSUE */
57     "Type is not an integer or float", /* ECTF_NOTINTFP */
58     "Type is not an array", /* ECTF_NOTARRAY */
59     "Type does not reference another type", /* ECTF_NOTREF */
60     "Input buffer is too small for type name", /* ECTF_NAMELEN */
61     "No type information available for that name", /* ECTF_NOTYPE */
62     "Syntax error in type name", /* ECTF_SYNTAX */
63     "Symbol table entry is not a function", /* ECTF_NOTFUNC */
64     "No function information available for symbol", /* ECTF_NOFUNC DAT */
65     "Symbol table entry is not a data object", /* ECTF_NOTDATA */
66     "No type information available for symbol", /* ECTF_NOTYPEDAT */
67     "No label information available for that name", /* ECTF_NOLABEL */
68     "File does not contain any labels", /* ECTF_NOLABELDATA */
69     "Feature not supported", /* ECTF_NOTSUP */
70     "Invalid enum element name", /* ECTF_NOENUMNAM */
71     "Invalid member name", /* ECTF_NOMEMNAM */
72     "CTF container is read-only", /* ECTF_RDONLY */
73     "Limit on number of dynamic type members reached", /* ECTF_DTFULL */
74     "Limit on number of dynamic types reached", /* ECTF_FULL */
75     "Duplicate member name definition", /* ECTF_DUPMEMBER */
76     "Conflicting type is already defined", /* ECTF_CONFLICT */
77     "Type has outstanding references", /* ECTF_REFERENCED */
78     "Type is not a dynamic type", /* ECTF_NOTDYN */
79     "Elf library failure", /* ECTF_ELF */
80     "Cannot merge child container", /* ECTF_MCHILD */
81     "Label already exists", /* ECTF_LABEL */
82     "Merged labels conflict", /* ECTF_LCONFLICT */
83     "Zlib library failure", /* ECTF_ZLIB */
84     "CTF conversion backend error", /* ECTF_CONVBKERR */
85     "No C source to convert from", /* ECTF_CONVNOCSRC */
86     "No debug info found to convert from" /* ECTF_CONVNODEBUG */
86     "No applicable conversion backend" /* ECTF_NOCONVBKEND */
87 };
      unchanged_portion_omitted

```

new/usr/src/lib/Makefile.lib

1

```
*****
8452 Thu Apr 25 02:20:41 2019
new/usr/src/lib/Makefile.lib
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 2019 OmniOS Community Edition (OmniOSce) Association.
22 # Copyright 2015 Gary Mills
23 # Copyright 2015 Igor Kozhukhov <ikozhukhov@gmail.com>
24 # Copyright (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
25 # Copyright 2019, Joyent, Inc.
26 # Copyright (c) 2015, Joyent, Inc.
27 #
28 # Definitions common to libraries.
29 #
30 # include global definitions; SRC should be defined in the shell.
31 # SRC is needed until RFE 1026993 is implemented.
32 #
33 include $(SRC)/Makefile.master
34 #
35 ORDER= lorder
36 TSORT= tsort
37 #
38 #
39 # By default, we define the source directory for libraries to be
40 # one level up from the ISA-specific directory, where the code is
41 # actually built. Many libraries define a 'common' directory to
42 # contain the source. These libraries must redefine SRCDIR as:
43 # SRCDIR = ../common
44 # Other variations are possible (../port, ../src, etc).
45 #
46 SRCDIR = ..
47 #
48 #
49 # We define MAPFILES here for the benefit of most libraries, those that
50 # follow the convention of having source files and other common files
51 # in the $(SRCDIR) directory. Libraries that do not follow this
52 # convention must define MAPFILES, or MAPFILEDIR for themselves.
```

new/usr/src/lib/Makefile.lib

2

```
53 # Libraries that do follow this convention but that need supplemental
54 # ISA-specific mapfiles can augment MAPFILES like this:
55 # MAPFILES += mapfile-vers
56 #
57 MAPFILEDIR = $(SRCDIR)
58 MAPFILES = $(MAPFILEDIR)/mapfile-vers
59 #
60 #
61 # If HDRDIR is left unset, then it's possible for the $(ROOTHDRDIR)/%
62 # install rule in lib/Makefile.targ to generate false matches if there
63 # are any common directory names between / and /usr/include ('xfn' is
64 # one common example). To prevent this, we set HDRDIR to a directory
65 # name that will almost surely not exist on the build machine.
66 #
67 HDRDIR= /__nonexistent_directory__
68 #
69 #
70 # We don't build archive (*.a) libraries by default anymore.
71 # If a component of the build needs to build an archive library
72 # for its own internal purposes, it can define LIBS for itself
73 # after including Makefile.lib, like this:
74 # LIBS = $(LIBRARY)
75 # or:
76 # LIBS = $(LIBRARYCCC)
77 # Archive libraries must not be installed in the proto area.
78 #
79 LIBS=
80 MACHLIBS= $(LIBS:%=$(MACH)/%)
81 MACHLIBS64= $(LIBS:%=$(MACH64)/%)
82 DYNLIB= $(LIBRARY:.a=.so$(VERS))
83 DYNLIBPSR= $(LIBRARY:.a=_psr.so$(VERS))
84 DYNLIBCCC= $(LIBRARYCCC:.a=.so$(VERS))
85 LIBLINKS= $(LIBRARY:.a=.so)
86 LIBLINKSCCC= $(LIBRARYCCC:.a=.so)
87 LIBNAME= $(LIBRARY:lib%.a=%)
88 LIBLINKPATH=
89 LIBNULL= null.a
90 ROOTHDRDIR= $(ROOT)/usr/include
91 ROOTLIBDIR= $(ROOT)/usr/lib
92 ROOTLIBDIR64= $(ROOT)/usr/lib/$(MACH64)
93 ROOTFS_LIBDIR= $(ROOT)/lib
94 ROOTFS_LIBDIR64= $(ROOT)/lib/$(MACH64)
95 ROOTHDRS= $(HDRS:%=$(ROOTHDRDIR)/%)
96 HDRSRCS= $(HDRS:%=$(HDRDIR)/%)
97 CHECKHDRS= $(HDRSRCS:%.h=%.check)
98 ROOTLIBS= $(LIBS:%=$(ROOTLIBDIR)/%)
99 ROOTLIBS64= $(LIBS:%=$(ROOTLIBDIR64)/%)
100 ROOTFS_LIBS= $(DYNLIB:%=$(ROOTFS_LIBDIR)/%)
101 ROOTFS_LIBS64= $(DYNLIB:%=$(ROOTFS_LIBDIR64)/%)
102 ROOTLINKS= $(ROOTLIBDIR)/$(LIBLINKS)
103 ROOTLINKS64= $(ROOTLIBDIR64)/$(LIBLINKS)
104 ROOTFS_LINKS= $(ROOTFS_LIBDIR)/$(LIBLINKS)
105 ROOTFS_LINKS64= $(ROOTFS_LIBDIR64)/$(LIBLINKS)
106 ROOTLINKSCCC= $(ROOTLIBDIR)/$(LIBLINKSCCC)
107 ROOTLINKSCCC64= $(ROOTLIBDIR64)/$(LIBLINKSCCC)
108 ROOTFS_LINKSCCC= $(ROOTFS_LIBDIR)/$(LIBLINKSCCC)
109 ROOTFS_LINKSCCC64= $(ROOTFS_LIBDIR64)/$(LIBLINKSCCC)
110 #
111 # Demo rules
112 DEMOFILES=
113 DEMOFILESRCDIR= common
114 ROOTDEMODIRBASE= __nonexistent_directory__
115 ROOTDEMODIRS=
116 ROOTDEMOFILES= $(DEMOFILES:%=$(ROOTDEMODIRBASE)/%)
117 $(ROOTDEMODIRS) := DIRMODE = 755
```

new/usr/src/lib/Makefile.lib

3

```

119 ARFLAGS=      r
120 SONAME=       $(DYNLIB)
121 # For most libraries, we should be able to resolve all symbols at link time,
122 # either within the library or as dependencies, all text should be pure, and
123 # combining relocations into one relocation table reduces startup costs.
124 # All options are tunable to allow overload/omission from lower makefiles.

127 HSONAME=      -h$(SONAME)
128 DYNFLAGS=     $(HSONAME) $(ZTEXT) $(ZDEFS) $(BDIRECT) \
129              $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-M%)

131 LDLIBS=       $(LDLIBS.lib)

133 OBJS=         $(OBJECTS:%=objs/%)
134 PICS=         $(OBJECTS:%=pics/%)

136 # Declare that all library .o's can all be made in parallel.
137 # The DUMMY target is for those instances where OBJS and PICS
138 # are empty (to avoid an unconditional .PARALLEL declaration).
139 .PARALLEL:    $(OBJS) $(PICS) DUMMY

141 # default value for "portable" source
142 SRCS=         $(OBJECTS:%.o=$(SRCDIR)/%.c)

144 # default build of an archive and a shared object,
145 # overridden locally when extra processing is needed
146 BUILD.AR=     $(AR) $(ARFLAGS) $@ $(AROBJ)
147 BUILD.SO=     $(CC) $(CFLAGS) -o $@ $(GSHARED) $(DYNFLAGS) \
148              $(PICS) $(EXTPICS) $(LDLIBS)
149 BUILD.CC.SO=  $(CCC) $(CCFLAGS) -o $@ $(GSHARED) $(DYNFLAGS) \
150              $(PICS) $(EXTPICS) $(LDLIBS)

152 # default dynamic library symlink
153 INS.liblink=  -$(RM) $@; $(SYMLINK) $(LIBLINKPATH)$(LIBLINKS)$(VERS) $@
154 INS.liblinkccc= -$(RM) $@; $(SYMLINK) $(LIBLINKPATH)$(LIBLINKSCC)$(VERS) $@

156 # default 64-bit dynamic library symlink
157 INS.liblink64= -$(RM) $@; $(SYMLINK) $(LIBLINKPATH)$(LIBLINKS)$(VERS) $@
158 INS.liblinkccc64= -$(RM) $@; $(SYMLINK) $(LIBLINKPATH)$(LIBLINKSCC)$(VERS) $@

160 #
161 # If appropriate, augment POST_PROCESS_O and POST_PROCESS_SO to do CTF
162 # processing. We'd like to just conditionally append to POST_PROCESS_O and
163 # POST_PROCESS_SO, but ParallelMake has a bug which causes the same value to
164 # sometimes get appended more than once, which will cause ctffconvert to fail.
165 # So, instead we introduce CTFCONVERT_POST and CTFMERGE_POST, which are always
166 # appended to POST_PROCESS_O and POST_PROCESS_SO but are no-ops unless CTF
167 # processing should be done.
168 #
169 CTFCONVERT_POST = :
170 CTFMERGE_POST = :
171 POST_PROCESS_O += ; $(CTFCONVERT_POST)
172 POST_PROCESS_SO += ; $(CTFMERGE_POST)

174 CTFMERGE_LIB = $(CTFMERGE) $(CTFMRGFLAGS) -t -f -L VERSION -o $@ $(PICS)

176 # conditional assignments

178 $(OBJ) :=      sparc_CFLAGS += -xregs=no%appl

180 $(PICS) :=     sparc_CFLAGS += -xregs=no%appl $(sparc_C_PICFLAGS)
181 $(PICS) :=     sparcv9_CFLAGS += -xregs=no%appl $(sparcv9_C_PICFLAGS)
182 $(PICS) :=     i386_CFLAGS += $(i386_C_PICFLAGS)
183 $(PICS) :=     amd64_CFLAGS += $(amd64_C_PICFLAGS)
184 $(PICS) :=     CCFLAGS += $(CC_PICFLAGS)

```

new/usr/src/lib/Makefile.lib

4

```

185 $(PICS) :=     CPPFLAGS += -DPIC -D_REENTRANT
186 $(PICS) :=     sparcv9_CCFLAGS += -xregs=no%appl $(sparcv9_CC_PICFLAGS)
187 $(PICS) :=     amd64_CCFLAGS += $(amd64_CC_PICFLAGS)
188 $(PICS) :=     CFLAGS += $(CTF_FLAGS)
189 $(PICS) :=     CFLAGS64 += $(CTF_FLAGS)
190 $(PICS) :=     CTFCONVERT_POST = $(CTFCONVERT_O)
191 $(DYNLIB) :=    CTFMERGE_POST = $(CTFMERGE_LIB)

193 $(LIBRARY):=   AROBJ = $(OBJ)
194 $(LIBRARY):=   DIR = objs
195 $(DYNLIB):=    DIR = pics
196 $(DYNLIBCCC):= DIR = pics

198 SONAMECCC=     $(DYNLIBCCC)
199 HSONAMECCC=    -h $(SONAMECCC)
200 #
201 # Keep in sync with the standard DYNFLAGS
202 #
203 $(DYNLIBCCC):=  DYNFLAGS = $(HSONAMECCC) $(ZTEXT) $(ZDEFS) \
204                $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-M%) \
205                $(BDIRECT) $(NORUNPATH)

208 # build rule for "portable" source
209 objs/%.o pics/%.o: %.c
210     $(COMPILE.c) -o $@ $<
211     $(POST_PROCESS_O)

213 objs/%.o pics/%.o: %.cc
214     $(COMPILE.cc) -o $@ $<
215     $(POST_PROCESS_CC_O)
215     $(POST_PROCESS_O)

217 .PRECIOUS: $(LIB)

219 # Define the majority text domain in this directory.
220 TEXT_DOMAIN= SUNW_OST_OSLIB

222 #
223 # Target Architecture
224 #
225 TARGETMACH=    $(MACH)

227 #
228 # Allow people to define their own clobber rules. Normal makefiles
229 # shouldn't override this - they should override $(CLOBBERFILES) instead.
230 #
231 CLOBBERTARGETFILES= $(LIB) $(DYNLIB) $(CLOBBERFILES)

233 #
234 # Define the default ctfdiff invocation used to check a list of types
235 # supplied by a user of a library. The goal is to validate that a given
236 # series of types is the same in both a 32-bit and 64-bit artifact. This
237 # is only defined if we have a 64-bit build to do.
238 #
239 TYPECHECK_LIB32 = $(TYPECHECK_LIB:%=$(MACH)/%)
240 TYPECHECK_LIB64 = $(TYPECHECK_LIB:%=$(MACH64)/%)
241 TYPECHECK_LIST=  $(TYPELIST:%=-T %)
242 $(BUILD64)TYPECHECK.lib = $(CTFDIFF) -t -I $(TYPECHECK_LIST) $(TYPECHECK_L
243 TYPECHECK =      $(TYPECHECK_LIB:%=%).typecheck

```

new/usr/src/lib/Makefile.targ

1

```
*****
2994 Thu Apr 25 02:20:41 2019
new/usr/src/lib/Makefile.targ
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright (c) 2019, Joyent, Inc.
24 #
25 #
26 #
27 # included to define local library targets
28 #
29 #
30 # conditional assignment of default permissions for the installed
31 # DYNLIB/DYNLIBBCC. Set here and in Makefile.mach, instead of Makefile.lib,
32 # because ROOTLIBDIR may be changed in the including makefile after it has
33 # included Makefile.lib.
34 #
35 $(ROOTLIBDIR)/$(DYNLIB) := FILEMODE= 755
36 $(ROOTLIBDIR)/$(DYNLIBBCC) := FILEMODE= 755
37 #
38 $(ROOTFS_LIBDIR)/$(DYNLIB) := FILEMODE= 755
39 $(ROOTFS_LIBDIR)/$(DYNLIBBCC) := FILEMODE= 755
40 #
41 $(ROOTLIBDIR64)/$(DYNLIB) := FILEMODE= 755
42 $(ROOTLIBDIR64)/$(DYNLIBBCC) := FILEMODE= 755
43 #
44 # install rules
45 $(ROTHDRDIR)/%: $(HDRDIR)/%
46     $(INS.file)
47 #
48 $(ROOTLIBDIR)/%: %
49     $(INS.file)
50 $(ROOTLIBDIR64)/%: %
51     $(INS.file)
52 $(ROOTFS_LIBDIR)/%: %
53     $(INS.file)
```

new/usr/src/lib/Makefile.targ

2

```
54 $(ROOTFS_LIBDIR64)/%: %
55     $(INS.file)
56 #
57 $(ROOTLIBDIR)/$(LIBLINKS): $(ROOTLIBDIR)/$(LIBLINKS)$(VERS)
58     $(INS.liblink)
59 $(ROOTLIBDIR64)/$(LIBLINKS): $(ROOTLIBDIR64)/$(LIBLINKS)$(VERS)
60     $(INS.liblink64)
61 #
62 $(ROOTLIBDIR)/$(LIBLINKSCCC): $(ROOTLIBDIR)/$(LIBLINKSCCC)$(VERS)
63     $(INS.liblinkccc)
64 $(ROOTLIBDIR64)/$(LIBLINKSCCC): $(ROOTLIBDIR64)/$(LIBLINKSCCC)$(VERS)
65     $(INS.liblinkccc64)
66 #
67 $(ROOTLINTDIR)/%: $(SRCDIR)/%
68     $(INS.file)
69 $(ROOTFS_LINTDIR)/%: $(SRCDIR)/%
70     $(INS.file)
71 #
72 $(ROOTDEMODIRS):
73     $(INS.dir)
74 $(ROOTDEMODIRBASE)/%: $(DEMOFILESRCDIR)/%
75     $(INS.file)
76 #
77 objs/%.o pics/%.o: $(SRCDIR)/%.c
78     $(COMPILE.c) -o $@ $<
79     $(POST_PROCESS_O)
80 #
81 objs/%.o pics/%.o: $(SRCDIR)/%.cc
82     $(COMPILE.cc) -o $@ $<
83     $(POST_PROCESS_CC_O)
84     $(POST_PROCESS_O)
85 #
86 objs pics:
87     -@mkdir -p $@
88 #
89 $(LIBRARY): objs .WAIT $$ (OBJS)
90     $(BUILD.AR)
91     $(POST_PROCESS_A)
92 #
93 $(DYNLIB): $$ (MAPFILES)
94 #
95 $(DYNLIB): pics .WAIT $$ (PICS) $$ (ALTPICS) $$ (EXTPICS)
96     $(BUILD.SO)
97     $(POST_PROCESS_SO)
98 #
99 $(DYNLIBBCC): pics .WAIT $$ (PICS) $$ (ALTPICS) $$ (EXTPICS)
100     $(BUILDCCC.SO)
101     $(POST_PROCESS_SO)
102 #
103 $(LINTLIB): $$ (SRCS)
104     $(LINT.c) -o $(LIBNAME) $(SRCS) > $(LINTOUT) 2>&1
105 #
106 lintcheck: $$ (SRCS)
107     $(LINT.c) $(LINTCHECKFLAGS) $(SRCS) $(LDLIBS)
108 #
109 $(TYPECHECK): $(TYPECHECK_LIB32) $(TYPECHECK_LIB64)
110     $(TYPECHECK.lib)
111 #
112 clobber: clean
113     -$(RM) $(CLOBBERTARGETFILES)
114 #
115 clean:
116     -$(RM) $(OBJS)
117     -$(RM) $(PICS) $(DUPLICATE_SRC) $(LINTOUT) $(LINTLIB) $(CLEANFILES)
```

new/usr/src/lib/brand/shared/brand/Makefile.com

1

```
*****
2530 Thu Apr 25 02:20:41 2019
new/usr/src/lib/brand/shared/brand/Makefile.com
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 (c) 2010, Oracle and/or its affiliates. All rights reserved.
23 #
24 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
25 #
26 # Copyright (c) 2019, Joyent, Inc.
27 #

29 COBJS = brand_util.o
30 ASOBSJS = crt.o handler.o runexe.o
31 OFFSETS_SRC = ../common/offsets.in
32 OFFSETS_H = assym.h
33 OBJECTS = $(COBJS) $(ASOBSJS)
34 CLOBBERFILES += $(OFFSETS_H)

36 include $(SRC)/lib/Makefile.lib

38 SRCDIR = ../common
39 CSRCS = $(COBJS:%o=../common/%c)
40 ASSRCS = $(ASOBSJS:%o=$(ISASRCDIR)/%s)
41 SRCS = $(CSRCS) $(ASSRCS)

43 #
44 # Ugh, this is a gross hack. Our assembly routines uses lots of defines
45 # to simplify variable access. All these defines work fine for amd64
46 # compiles because when compiling for amd64 we use the GNU assembler,
47 # gas. For 32-bit code we use the Sun assembler, as. Unfortunatly
48 # as does not handle certian constructs that gas does. So rather than
49 # make our code less readable, we'll just use gas to compile our 32-bit
50 # code as well.
51 #
52 i386_AS = $(amd64_AS)
```

new/usr/src/lib/brand/shared/brand/Makefile.com

2

```
54 CPPFLAGS += -D_REENTRANT -U_ASM -I. -I../sys
55 CFLAGS += $(CVERBOSE)
56 ASFLAGS = -P $(ASFLAGS_$(CURTYPE)) -D_ASM -I. -I../sys

58 .KEEP_STATE:

60 #
61 # build the offset header before trying to compile any files. (it's included
62 # by brand_misc.h, so it's needed for all objects, not just assembly ones.)
63 #
64 # Note we have to build assym.h via its dependency on pics/% so that the
65 # target dependent assignment of CTF_FLAGS will be there, otherwise make
66 # will see two different commands to build it (endless rebuilds).
67 #
68 all: pics .WAIT $$ (PICS)

70 lint: lintcheck

72 $(OBJECTS:%=pics/%): $(OFFSETS_H)

74 $(OFFSETS_H): $(OFFSETS_SRC)
75     $(OFFSETS_CREATE) < $(OFFSETS_SRC) >$$

77 pics/%.o: $(ISASRCDIR)/%.s
78     $(COMPILE.s) -o $$@ $$<
79     $(POST_PROCESS_S_O)
77     $(POST_PROCESS_O)

81 include $(SRC)/lib/Makefile.targ
```

new/usr/src/lib/compmpage/Makefile.shared.targ

1

807 Thu Apr 25 02:20:42 2019

new/usr/src/lib/compmpage/Makefile.shared.targ

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

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 at

9 # <http://www.illumos.org/license/CDDL>.

10 #

12 #

13 # Copyright (c) 2019, Joyent, Inc.

13 # Copyright 2016 Joyent, Inc.

14 #

16 #

17 # This Makefile is shared between both libc and other consumers

18 #

20 pics/%.o: \$(SRC)/lib/compmpage/common/%.c

21 \$(COMPILE.c) -o \$@ \$<

22 \$(POST_PROCESS_O)

24 pics/%.o: \$(SRC)/lib/compmpage/\$(TARGET_ARCH)/%.s \$(COMPMPAGE_OFFSETS_H)

25 \$(COMPILE.s) -o \$@ \$<

26 \$(POST_PROCESS_S_O)

26 \$(POST_PROCESS_O)

28 \$(COMPMPAGE_OFFSETS_H): \$(COMPMPAGE_OFFSETS_SRC)

29 \$(OFFSETS_CREATE) <\$(COMPMPAGE_OFFSETS_SRC) >\$@

new/usr/src/lib/libc/Makefile.targ

1

7457 Thu Apr 25 02:20:42 2019
new/usr/src/lib/libc/Makefile.targ
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 # Copyright 2016 Joyent, Inc.
24 #
25 # Copyright (c) 2019, Joyent, Inc.
26 #

28 # libc build rules

30 #
31 # This first rule picks up optimized sources for libc variants.
32 #
33 pics%.o: $(OPTIMIZED_LIBCBASE)/gen%.s
34     $(BUILD.s)
35     $(POST_PROCESS_S_O)
35     $(POST_PROCESS_O)

37 pics%.o: $(LIBCBASE)/crt%.c
38     $(COMPILE.c) -o $@ $<
39     $(POST_PROCESS_O)

41 pics%.o: $(LIBCBASE)/crt%.s
42     $(BUILD.s)
43     $(POST_PROCESS_S_O)
43     $(POST_PROCESS_O)

45 pics%.o: $(LIBCBASE)/gen%.c
46     $(COMPILE.c) -o $@ $<
47     $(POST_PROCESS_O)

49 pics%.o: $(LIBCBASE)/gen%.s
50     $(BUILD.s)
```

new/usr/src/lib/libc/Makefile.targ

2

```
51     $(POST_PROCESS_S_O)
51     $(POST_PROCESS_O)

53 pics%.o: $(LIBCBASE)/../$(TARGET_ARCH)/gen%.s
54     $(BUILD.s)
55     $(POST_PROCESS_S_O)
55     $(POST_PROCESS_O)

57 pics/_stack_grow.o: $(LIBCBASE)/../$(TARGET_ARCH)/gen/_stack_grow.s
58     $(BUILD.s) $(LIBCBASE)/../$(TARGET_ARCH)/gen/_stack_grow.s
59     $(POST_PROCESS_S_O)
59     $(POST_PROCESS_O)

61 pics%.o: $(LIBCBASE)/fp%.c
62     $(COMPILE.c) -o $@ $<
63     $(POST_PROCESS_O)

65 pics%.o: $(LIBCBASE)/../$(MACH)/fp%.c
66     $(COMPILE.c) -o $@ $<
67     $(POST_PROCESS_O)

69 pics%.o: $(LIBCBASE)/fp%.s
70     $(BUILD.s)
71     $(POST_PROCESS_S_O)
71     $(POST_PROCESS_O)

73 pics%.o: $(LIBCBASE)/../$(TARGET_ARCH)/fp%.c
74     $(COMPILE.c) -o $@ $<
75     $(POST_PROCESS_O)

77 pics%.o: $(LIBCBASE)/../$(TARGET_ARCH)/fp%.s
78     $(BUILD.s)
79     $(POST_PROCESS_S_O)
79     $(POST_PROCESS_O)

81 pics%.o: $(LIBCBASE)/sys%.c
82     $(COMPILE.c) -o $@ $<
83     $(POST_PROCESS_O)

85 pics%.o: $(LIBCBASE)/../common/sys%.s
86     $(BUILD.s)
87     $(POST_PROCESS_S_O)
87     $(POST_PROCESS_O)

89 pics%.o: $(LIBCBASE)/sys%.s
90     $(BUILD.s)
91     $(POST_PROCESS_S_O)
91     $(POST_PROCESS_O)

93 pics%.o: $(LIBCBASE)/../$(MACH)/sys%.s
94     $(BUILD.s)
95     $(POST_PROCESS_S_O)
95     $(POST_PROCESS_O)

97 pics%.o: $(LIBCBASE)/../$(MACH)/sys%.c
98     $(COMPILE.c) -o $@ $<
99     $(POST_PROCESS_O)

101 pics%.o: $(LIBCBASE)/../$(TARGET_ARCH)/sys%.c
102     $(COMPILE.c) -o $@ $<
103     $(POST_PROCESS_O)

105 pics%.o: $(LIBCBASE)/../$(TARGET_ARCH)/sys%.s
106     $(BUILD.s)
107     $(POST_PROCESS_S_O)
107     $(POST_PROCESS_O)
```

```

109 # rules to build large file aware objects (xxx64.o from xxx.s or xxx.c)

111 pics/%64.o: $(LIBCBASE)/../common/sys/%.s
112     $(BUILD.s)
113     $(POST_PROCESS_S_O)
113     $(POST_PROCESS_O)

115 pics/%64.o: $(LIBCBASE)/sys/%.s
116     $(BUILD.s)
117     $(POST_PROCESS_S_O)
117     $(POST_PROCESS_O)

119 pics/%64.o: $(LIBCBASE)/gen/%.c
120     $(COMPILE.c) -o $@ $<
121     $(POST_PROCESS_O)

123 pics/%64.o: $(LIBCBASE)/../port/gen/%.c
124     $(COMPILE.c) -o $@ $<
125     $(POST_PROCESS_O)

127 pics/%64.o: $(LIBCBASE)/../port/sys/%.c
128     $(COMPILE.c) -o $@ $<
129     $(POST_PROCESS_O)

131 pics/%64.o: $(LIBCBASE)/../port/print/%.c
132     $(COMPILE.c) -o $@ $<
133     $(POST_PROCESS_O)

135 pics/%64.o: $(LIBCBASE)/../port/regex/%.c
136     $(COMPILE.c) -DM_I18N_MB -DI18N \
137     -I$(LIBCBASE)/../port/regex \
138     -I$(LIBCBASE)/../port/gen -o $@ $<
139     $(POST_PROCESS_O)

141 pics/%64.o: $(LIBCBASE)/../port/stdio/%.c
142     $(COMPILE.c) -o $@ $<
143     $(POST_PROCESS_O)

145 pics/%_w.o: $(LIBCBASE)/../port/stdio/%.c
146     $(COMPILE.c) -o $@ $<
147     $(POST_PROCESS_O)

149 pics/%_w.o: $(LIBCBASE)/../port/print/%.c
150     $(COMPILE.c) -o $@ $<
151     $(POST_PROCESS_O)

153 pics/%_pos.o: $(LIBCBASE)/../port/il8n/%.c
154     $(COMPILE.c) -o $@ $<
155     $(POST_PROCESS_O)

157 pics/%_sbyte.o: $(LIBCBASE)/../port/il8n/%.c
158     $(COMPILE.c) -o $@ $<
159     $(POST_PROCESS_O)

161 pics/%_possbyte.o: $(LIBCBASE)/../port/il8n/%.c
162     $(COMPILE.c) -o $@ $<
163     $(POST_PROCESS_O)

165 pics/%_longlong.o: $(LIBCBASE)/../port/il8n/%.c
166     $(COMPILE.c) -o $@ $<
167     $(POST_PROCESS_O)

169 # libc build rules for objects built from "portable" source in ../port

171 pics/%.o: $(LIBCBASE)/../port/fp/%.c

```

```

172     $(COMPILE.c) \
173     -I$(LIBCBASE)/../port/fp -o $@ $<
174     $(POST_PROCESS_O)

176 pics/%.o: $(LIBCBASE)/../port/il8n/%.c
177     $(COMPILE.c) \
178     -I$(LIBCBASE)/../port/il8n -o $@ $<
179     $(POST_PROCESS_O)

181 # gen rules
182 pics/%.o %.o: $(LIBCBASE)/../port/gen/%.c
183     $(COMPILE.c) -o $@ $<
184     $(POST_PROCESS_O)

186 # inet rules
187 pics/%.o %.o: $(LIBCBASE)/../port/inet/%.c
188     $(COMPILE.c) -o $@ $<
189     $(POST_PROCESS_O)

191 # locale rules
192 pics/%.o %.o: $(LIBCBASE)/../port/locale/%.c
193     $(COMPILE.c) -o $@ $<
194     $(POST_PROCESS_O)

196 # print rules
197 pics/%.o: $(LIBCBASE)/../port/print/%.c
198     $(COMPILE.c) -o $@ $<
199     $(POST_PROCESS_O)

201 # regex rules
202 pics/%.o: $(LIBCBASE)/../port/regex/%.c
203     $(COMPILE.c) -DM_I18N_MB -DI18N \
204     -I$(LIBCBASE)/../port/regex \
205     -I$(LIBCBASE)/../port/gen -o $@ $<
206     $(POST_PROCESS_O)

208 # stdio rules
209 pics/%.o: $(LIBCBASE)/../port/stdio/%.c
210     $(COMPILE.c) -o $@ $<
211     $(POST_PROCESS_O)

213 # c89 print, stdio rules
214 pics/%_c89.o: $(LIBCBASE)/../port/print/%.c
215     $(COMPILE.c) -o $@ $<
216     $(POST_PROCESS_O)

218 pics/%_c89.o: $(LIBCBASE)/../port/stdio/%.c
219     $(COMPILE.c) -o $@ $<
220     $(POST_PROCESS_O)

222 # aio rules
223 pics/%.o: $(LIBCBASE)/../port/aio/%.c
224     $(COMPILE.c) -o $@ $<
225     $(POST_PROCESS_O)

227 # rt rules
228 pics/%.o: $(LIBCBASE)/../port/rt/%.c
229     $(COMPILE.c) -o $@ $<
230     $(POST_PROCESS_O)

232 # tpool rules
233 pics/%.o: $(LIBCBASE)/../port/tpool/%.c
234     $(COMPILE.c) -o $@ $<
235     $(POST_PROCESS_O)

237 # threads rules

```


new/usr/src/lib/libc/Makefile.targ

5

```

238 pics/%.o: $(LIBCBASE)/../port/threads/%.c
239     $(COMPILE.c) -o $@ $<
240     $(POST_PROCESS_O)

242 pics/%.o: $(LIBCBASE)/threads/%.c
243     $(COMPILE.c) -o $@ $<
244     $(POST_PROCESS_O)

246 pics/%.o: $(LIBCBASE)/threads/%.s
247     $(BUILD.s)
248     $(POST_PROCESS_S_O)
248     $(POST_PROCESS_O)

250 pics/%.o: $(LIBCBASE)/../$(TARGET_ARCH)/threads/%.c
251     $(COMPILE.c) -o $@ $<
252     $(POST_PROCESS_O)

254 pics/%.o: $(LIBCBASE)/../$(TARGET_ARCH)/threads/%.s
255     $(BUILD.s)
256     $(POST_PROCESS_S_O)
256     $(POST_PROCESS_O)

258 pics/%.o: $(LIBCBASE)/../$(TARGET_ARCH)/unwind/%.c
259     $(COMPILE.c) -o $@ $<
260     $(POST_PROCESS_O)

262 pics/%.o: $(LIBCBASE)/../$(TARGET_ARCH)/unwind/%.s
263     $(BUILD.s)
264     $(POST_PROCESS_S_O)
264     $(POST_PROCESS_O)

266 pics/%.o: $(LIBCBASE)/../port/unwind/%.c
267     $(COMPILE.c) -o $@ $<
268     $(POST_PROCESS_O)

270 pics/%.o: $(LIBCBASE)/../$(MACH)/unwind/%.s
271     $(BUILD.s)
272     $(POST_PROCESS_S_O)
272     $(POST_PROCESS_O)

274 pics/%.o: $(LIBCBASE)/../port/sys/%.c
275     $(COMPILE.c) -o $@ $<
276     $(POST_PROCESS_O)

278 pics/%.o: $(LIBCBASE)/../../crt/common/%.c
279     $(COMPILE.c) -o $@ $<
280     $(POST_PROCESS_O)

282 pics/%.o: $(LIBCBASE)/$(CRTSRCS)/%.s
283     $(BUILD.s)
284     $(POST_PROCESS_S_O)
284     $(POST_PROCESS_O)

286 # $(SRC)/common rules
287 pics/%.o: $(SRC)/common/atomic/$(TARGETMACH)/%.s
288     $(BUILD.s)
289     $(POST_PROCESS_S_O)
289     $(POST_PROCESS_O)

291 $(COMOBSJS:=pics/%): $(SRC)/common/util/$(@F:.o=.c)
292     $(COMPILE.c) -o $@ $(SRC)/common/util/$(@F:.o=.c)
293     $(POST_PROCESS_O)

295 $(XATTROBSJS:=pics/%): $(SRC)/common/xattr/$(@F:.o=.c)
296     $(COMPILE.c) -o $@ $(SRC)/common/xattr/$(@F:.o=.c)
297     $(POST_PROCESS_O)

```

new/usr/src/lib/libc/Makefile.targ

6

```

299 $(DTRACEOBSJS:=pics/%): $(SRC)/common/dtrace/$(@F:.o=.c)
300     $(COMPILE.c) -o $@ $(SRC)/common/dtrace/$(@F:.o=.c)
301     $(POST_PROCESS_O)

303 $(SECFLAGSOBSJS:=pics/%): $(SRC)/common/secflags/$(@F:.o=.c)
304     $(COMPILE.c) -o $@ $(SRC)/common/secflags/$(@F:.o=.c)
305     $(POST_PROCESS_O)

307 $(UNICODEOBSJS:=pics/%): $(SRC)/common/unicode/$(@F:.o=.c)
308     $(COMPILE.c) -o $@ $(SRC)/common/unicode/$(@F:.o=.c)
309     $(POST_PROCESS_O)

311 $(CHACHAOBSJS:=pics/%): $(SRC)/common/crypto/chacha/$(@F:.o=.c)
312     $(COMPILE.c) -I$(SRC)/common/crypto/chacha -DKEYSTREAM_ONLY \
313     -o $@ $(SRC)/common/crypto/chacha/$(@F:.o=.c)
314     $(POST_PROCESS_O)

316 # DTrace rules
317 pics/%.o: $(LIBCBASE)/../port/threads/%.d $(THREADSOBSJS:=pics/%)
318     $(COMPILE.d) -C -xlazyload -s $< -o $@ $(THREADSOBSJS:=pics/%)
319     $(POST_PROCESS_O)

321 include $(SRC)/lib/Makefile.targ

```

new/usr/src/lib/libc/capabilities/Makefile.targ

1

```
*****
1613 Thu Apr 25 02:20:42 2019
new/usr/src/lib/libc/capabilities/Makefile.targ
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
```

```
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 #
23 # Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
24 #
25 # Copyright (c) 2019, Joyent, Inc.
26 #
27 #
28 $(SYMCAP):      $(OBJCAP)
29 $(OBJCAP):      $(PICS)
30 $(PICS):        pics
31 #
32 # Assemble the original source file.
33 #
34 pics/%.o:       ../../$(TRG_PLATFORM)/common/%.s
35                 $(COMPILE.s) $< -o $@
36                 $(POST_PROCESS_S_O)
37                 $(POST_PROCESS_O)
38 #
39 pics/%.o:       ../../$(ALT_PLATFORM)/common/%.s
40                 $(COMPILE.s) $< -o $@
41                 $(POST_PROCESS_S_O)
42                 $(POST_PROCESS_O)
43 #
44 # Combine all pic objects into one relocatable object.  Assign any capabilities
45 # to this object, and define the interface.
46 #
47 pics/objcap.o:  $(PICS) $(MAPFILE-CAP)
48                 $(LD) -r -o $@ $(MAPOPT-CAP) -Breduce $(PICS)
49                 $(POST_PROCESS_OBJCAP_O)
50 #
51 # Convert the combined object capabilities object into a symbol capabilities
52 # object.
```

new/usr/src/lib/libc/capabilities/Makefile.targ

2

```
52 pics/symcap.o: $(OBJCAP)
53                 $(LD) -r -o $@ -z symbolcap $(OBJCAP)
54 #
55 include         ../../../../Makefile.targ
```

new/usr/src/lib/libc_db/Makefile.com

1

```
*****
1598 Thu Apr 25 02:20:42 2019
new/usr/src/lib/libc_db/Makefile.com
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
```

```
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 2008 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
25 # Copyright (c) 2019, Joyent, Inc.
25 # Copyright (c) 2018, Joyent, Inc.
```

```
27 LIBRARY = libc_db.a
28 VERS = .1
```

```
30 CRTI=  crt.i.o
31 CRTN=  crtn.o
```

```
33 CMNOBJS = thread_db.o
34 OBJECTS = $(CRTI) $(CMNOBJS) $(CRTN)
```

```
36 include .././Makefile.lib
37 include .././Makefile.rootfs
```

```
39 LIBS = $(DYNLIB) $(LINTLIB)
```

```
41 SRCDIR =      ../common
42 SRCS = $(CMNOBJS:%.o=$(SRCDIR)/%.c)
43 $(LINTLIB) := SRCS = $(SRCDIR)/$(LINTSRC)
```

```
45 ASFLAGS +=    -P -D__STDC__ -D_ASM -DPIC
46 CPPFLAGS +=   -I.././libc/inc -D_REENTRANT
47 CFLAGS +=     $(CCVERBOSE)
48 LDLIBS +=     -lc
```

```
50 CERRWARN +=   -_gcc=-Wno-type-limits
51 CERRWARN +=   -_gcc=-Wno-uninitialized
```

new/usr/src/lib/libc_db/Makefile.com

2

```
53 # not linted
54 SMATCH=off

56 .KEEP_STATE:

58 all: $(LIBS)

60 lint: lintcheck

62 include .././Makefile.targ

64 pics/%.o: $(CRTSRCS)/%.s
65     $(COMPILE.s) -o $@ $<
66     $(POST_PROCESS_S_O)
66     $(POST_PROCESS_O)
```

```

*****
1728 Thu Apr 25 02:20:43 2019
new/usr/src/lib/libcpc/Makefile.com
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****

```

```

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 (c) 1999, 2010, Oracle and/or its affiliates. All rights reserved.
23 #
24 # Copyright (c) 2019, Joyent, Inc.
24 # Copyright (c) 2018, Joyent, Inc.

```

```

26 LIBRARY = libcpc.a
27 VERS     = .1
28 COBJS    = libcpc.o subr.o
29 V1_OBJS  = obsoleted.o
30 OBJECTS  = $(ASOBJS) $(MACHCOBJS) $(COBJS) $(V1_OBJS)

```

```
32 include ../../Makefile.lib
```

```

34 SRCS = $(ASOBJS:%.o=./$(MACH)/%.s) \
35         $(MACHCOBJS:%.o=./$(MACH)/%.c) \
36         $(V1_OBJS:%.o=./common/%.c) \
37         $(COBJS:%.o=./common/%.c)

```

```

39 LIBS = $(DYNLIB) $(LINTLIB)
40 $(LINTLIB) := SRCS = ../common/lib-libcpc
41 LDLIBS += -lpctx -lnvpair -lc

```

```
43 SRCDIR = ../common
```

```

45 ASFLAGS += -P -D_ASM -I../common
46 CPPFLAGS += -D_REENTRANT -I../common
47 CFLAGS += $(CVERBOSE)

```

```

49 CERRWARN += -_gcc=-Wno-switch
50 CERRWARN += -_gcc=-Wno-uninitialized

```

```
52 # not linted
```

```

53 SMATCH=off
54
55 .KEEP_STATE:
56
57 all: $(LIBS)
58
59 lint: lintcheck
60
61 include ../../Makefile.targ
62
63 pics/%.o: ../$(MACH)/%.c
64         $(COMPILE.c) -o $@ $<
65         $(POST_PROCESS_O)
66
67 pics/%.o: ../$(MACH)/%.s
68         $(COMPILE.s) -o $@ $<
69         $(POST_PROCESS_S_O)
69         $(POST_PROCESS_O)

```

```

*****
4135 Thu Apr 25 02:20:43 2019
new/usr/src/lib/libctf/common/ctf_convert.c
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 at
9  * http://www.illumos.org/license/CDDL.
10 */

12 /*
13  * Copyright 2019 Joyent, Inc.
14 */

16 /*
17  * Main conversion entry points. This has been designed such that there can be
18  * any number of different conversion backends. Currently we only have one that
19  * understands DWARFv2 (and bits of DWARFv4). Each backend should be placed in
20  * the ctf_converters list and each will be tried in turn.
21 */

23 #include <libctf_impl.h>
24 #include <assert.h>
25 #include <gelf.h>

27 ctf_convert_f ctf_converters[] = {
28     ctf_dwarf_convert
29 };

31 #define NCONVERTS      (sizeof (ctf_converters) / sizeof (ctf_convert_f))

33 ctf_hsc_ret_t
34 ctf_has_c_source(Elf *elf, char *errmsg, size_t errlen)
35 {
36     typedef enum ctf_convert_source {
37         CTFCONV_SOURCE_NONE = 0x0,
38         CTFCONV_SOURCE_UNKNOWN = 0x01,
39         CTFCONV_SOURCE_C = 0x02,
40         CTFCONV_SOURCE_S = 0x04
41     } ctf_convert_source_t;

42     static void
43     ctf_convert_ftypes(Elf *elf, ctf_convert_source_t *types)
44     {
45         ctf_hsc_ret_t ret = CHR_NO_C_SOURCE;
46         Elf_Scn *scn, *strscn;
47         Elf_Data *data, *strdata;
48         int i;
49         Elf_Scn *scn = NULL, *strscn;
50         *types = CTFCONV_SOURCE_NONE;
51         GElf_Shdr shdr;
52         ulong_t i;
53         Elf_Data *data, *strdata;

```

```

42     scn = NULL;
43     while ((scn = elf_nextscn(elf, scn)) != NULL) {
44         if (gelf_getshdr(scn, &shdr) == NULL) {
45             (void) snprintf(errmsg, errlen,
46                 "failed to get section header: %s",
47                 elf_errmsg(elf_errno()));
48             return (CHR_ERROR);
49         }

50         if (gelf_getshdr(scn, &shdr) == NULL)
51             return;

52         if (shdr.sh_type == SHT_SYMTAB)
53             break;

54         if (scn == NULL)
55             return (CHR_NO_C_SOURCE);
56         return;

57         if ((strscn = elf_getscn(elf, shdr.sh_link)) == NULL) {
58             (void) snprintf(errmsg, errlen, "failed to get str section: %s",
59                 elf_errmsg(elf_errno()));
60             return (CHR_ERROR);
61         }
62         if ((strscn = elf_getscn(elf, shdr.sh_link)) == NULL)
63             return;

64         if ((data = elf_getdata(scn, NULL)) == NULL) {
65             (void) snprintf(errmsg, errlen, "failed to read section: %s",
66                 elf_errmsg(elf_errno()));
67             return (CHR_ERROR);
68         }
69         if ((data = elf_getdata(scn, NULL)) == NULL)
70             return;

71         if ((strdata = elf_getdata(strscn, NULL)) == NULL) {
72             (void) snprintf(errmsg, errlen,
73                 "failed to read string table: %s", elf_errmsg(elf_errno()));
74             return (CHR_ERROR);
75         }
76         if ((strdata = elf_getdata(strscn, NULL)) == NULL)
77             return;

78         for (i = 0; i < shdr.sh_size / shdr.sh_entsize; i++) {
79             GElf_Sym sym;
80             const char *file;
81             size_t len;

82             if (gelf_getsym(data, i, &sym) == NULL) {
83                 (void) snprintf(errmsg, errlen,
84                     "failed to read sym %lu: %s",
85                     i, elf_errmsg(elf_errno()));
86                 return (CHR_ERROR);
87             }
88             if (gelf_getsym(data, i, &sym) == NULL)
89                 return;

90             if (GELF_ST_TYPE(sym.st_info) != STT_FILE)
91                 continue;

92             file = (const char *)((uintptr_t)strdata->d_buf + sym.st_name);
93             len = strlen(file);
94             if (len >= 2 && strncmp(".c", &file[len - 2], 2) == 0) {
95                 ret = CHR_HAS_C_SOURCE;

```

```

82     if (len < 2 || file[len - 2] != '.') {
83         *types |= CTFCONV_SOURCE_UNKNOWN;
84         continue;
85     }

87     switch (file[len - 1]) {
88     case 'c':
89         *types |= CTFCONV_SOURCE_C;
90         break;
91     case 'h':
92         /* We traditionally ignore header files... */
93         break;
94     case 's':
95         *types |= CTFCONV_SOURCE_S;
96         break;
97     default:
98         *types |= CTFCONV_SOURCE_UNKNOWN;
99         break;
100    }

102    return (ret);
103 }

104 static ctf_file_t *
105 ctf_elfconvert(int fd, Elf *elf, const char *label, uint_t nthrs, uint_t flags,
106               int *errp, char *errbuf, size_t errlen)
107 {
108     int err, i;
109     ctf_file_t *fp = NULL;
110     boolean_t notsup = B_TRUE;
111     ctf_convert_source_t type;

112     if (errp == NULL)
113         errp = &err;

114     if (elf == NULL) {
115         *errp = EINVAL;
116         return (NULL);
117     }

118     if (flags & ~CTF_ALLOW_MISSING_DEBUG) {
119         if (flags & ~CTF_CONVERT_F_IGNNONC) {
120             *errp = EINVAL;
121             return (NULL);
122         }
123     }

124     if (elf_kind(elf) != ELF_K_ELF) {
125         *errp = ECTF_FMT;
126         return (NULL);
127     }

128     switch (ctf_has_c_source(elf, errbuf, errlen)) {
129     case CHR_ERROR:
130         *errp = ECTF_ELF;
131         return (NULL);

132     case CHR_NO_C_SOURCE:
133         ctf_convert_ftypes(elf, &type);
134         ctf_dprintf("got types: %d\n", type);
135         if (flags & CTF_CONVERT_F_IGNNONC) {
136             if (type == CTFCONV_SOURCE_NONE ||
137                 (type & CTFCONV_SOURCE_UNKNOWN)) {
138                 *errp = ECTF_CONVNOCSRC;
139                 return (NULL);
140             }
141         }
142     }

```

```

136     default:
137         break;
138     }
139 }

140     for (i = 0; i < NCONVERTS; i++) {
141         fp = NULL;
142         err = ctf_converters[i](fd, elf, nthrs, flags,
143                               &fp, errbuf, errlen);
144         ctf_conv_status_t cs;

145         if (err != ECTF_CONVNODEBUG)
146             fp = NULL;
147         cs = ctf_converters[i](fd, elf, nthrs, errp, &fp, errbuf,
148                               errlen);
149         if (cs == CTF_CONV_SUCCESS) {
150             notsup = B_FALSE;
151             break;
152         }

153         if (cs == CTF_CONV_ERROR) {
154             fp = NULL;
155             notsup = B_FALSE;
156             break;
157         }

158     }

159     if (err != 0) {
160         assert(fp == NULL);
161         *errp = err;
162         if (notsup == B_TRUE) {
163             if ((flags & CTF_CONVERT_F_IGNNONC) != 0 &&
164                 (type & CTFCONV_SOURCE_C) == 0) {
165                 *errp = ECTF_CONVNOCSRC;
166                 return (NULL);
167             }
168             *errp = ECTF_NOCONVBKEND;
169             return (NULL);
170         }
171     }

172     if (label != NULL) {
173         /*
174          * Successful conversion.
175          */
176         if (fp != NULL && label != NULL) {
177             if (ctf_add_label(fp, label, fp->ctf_ttypemax, 0) == CTF_ERR) {
178                 *errp = ctf_errno(fp);
179                 ctf_close(fp);
180                 return (NULL);
181             }
182             if (ctf_update(fp) == CTF_ERR) {
183                 *errp = ctf_errno(fp);
184                 ctf_close(fp);
185                 return (NULL);
186             }
187         }
188     }

189     return (fp);
190 }

```

unchanged portion omitted

84704 Thu Apr 25 02:20:43 2019

new/usr/src/lib/libctf/common/ctf_dwarf.c

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

unchanged portion omitted

```

843 static int
844 ctf_dwarf_dwarf_base(ctf_cu_t *cup, Dwarf_Die die, int *kindp,
845     ctf_encoding_t *enc)
846 {
847     int ret;
848     Dwarf_Signed type;

850     if ((ret = ctf_dwarf_signed(cup, die, DW_AT_encoding, &type)) != 0)
851         return (ret);

853     switch (type) {
854     case DW_ATE_unsigned:
855     case DW_ATE_address:
856         *kindp = CTF_K_INTEGER;
857         enc->cte_format = 0;
858         break;
859     case DW_ATE_unsigned_char:
860         *kindp = CTF_K_INTEGER;
861         enc->cte_format = CTF_INT_CHAR;
862         break;
863     case DW_ATE_signed:
864         *kindp = CTF_K_INTEGER;
865         enc->cte_format = CTF_INT_SIGNED;
866         break;
867     case DW_ATE_signed_char:
868         *kindp = CTF_K_INTEGER;
869         enc->cte_format = CTF_INT_SIGNED | CTF_INT_CHAR;
870         break;
871     case DW_ATE_boolean:
872         *kindp = CTF_K_INTEGER;
873         enc->cte_format = CTF_INT_SIGNED | CTF_INT_BOOL;
874         break;
875     case DW_ATE_float:
876     case DW_ATE_complex_float:
877     case DW_ATE_imaginary_float:
878     case DW_ATE_SUN_imaginary_float:
879     case DW_ATE_SUN_interval_float:
880         *kindp = CTF_K_FLOAT;
881         if ((ret = ctf_dwarf_float_base(cup, type, enc)) != 0)
882             return (ret);
883         break;
884     default:
885         (void) snprintf(cup->cu_errbuf, cup->cu_errlen,
886             "encountered unknown DWARF encoding: %d", type);
887         (void) snprintf(cup->cu_errbuf, cup->cu_errlen,
888             "encountered unknown DWARF encoding: %d", type);
889         return (ECTF_CONVBKERR);
890     }
891 }

```

unchanged portion omitted

```

1691 static int
1692 ctf_dwarf_convert_type(ctf_cu_t *cup, Dwarf_Die die, ctf_id_t *idp,
1693     int isroot)
1694 {
1695     int ret;
1696     Dwarf_Off offset;
1697     Dwarf_Half tag;
1698     ctf_dwmap_t lookup, *map;
1699     ctf_id_t id;

1701     if (idp == NULL)
1702         idp = &id;

1704     if ((ret = ctf_dwarf_offset(cup, die, &offset)) != 0)
1705         return (ret);

1707     if (offset > cup->cu_maxoff) {
1708         (void) snprintf(cup->cu_errbuf, cup->cu_errlen,
1709             "die offset %llu beyond maximum for header %llu\n",
1710             offset, cup->cu_maxoff);
1711         return (ECTF_CONVBKERR);
1712     }

1714     /*
1715      * If we've already added an entry for this offset, then we're done.
1716      */
1717     lookup.cdm_off = offset;
1718     if ((map = avl_find(&cup->cu_map, &lookup, NULL)) != NULL) {
1719         *idp = map->cdm_id;
1720         return (0);
1721     }

1723     if ((ret = ctf_dwarf_tag(cup, die, &tag)) != 0)
1724         return (ret);

1726     ret = ENOTSUP;
1727     switch (tag) {
1728     case DW_TAG_base_type:
1729         ctf_dprintf("base\n");
1730         ret = ctf_dwarf_create_base(cup, die, idp, isroot, offset);
1731         break;
1732     case DW_TAG_array_type:
1733         ctf_dprintf("array\n");
1734         ret = ctf_dwarf_create_array(cup, die, idp, isroot);
1735         break;
1736     case DW_TAG_enumeration_type:
1737         ctf_dprintf("enum\n");
1738         ret = ctf_dwarf_create_enum(cup, die, idp, isroot);
1739         break;
1740     case DW_TAG_pointer_type:
1741         ctf_dprintf("pointer\n");
1742         ret = ctf_dwarf_create_reference(cup, die, idp, CTF_K_POINTER,
1743             isroot);
1744         break;
1745     case DW_TAG_structure_type:
1746         ctf_dprintf("struct\n");
1747         ret = ctf_dwarf_create_sou(cup, die, idp, CTF_K_STRUCT,
1748             isroot);
1749         break;
1750     case DW_TAG_subroutine_type:
1751         ctf_dprintf("fptr\n");
1752         ret = ctf_dwarf_create_fptr(cup, die, idp, isroot);
1753         break;
1754     case DW_TAG_typedef:
1755         ctf_dprintf("typedef\n");

```

```

1756         ret = ctf_dwarf_create_reference(cup, die, idp, CTF_K_TYPEDEF,
1757         isroot);
1758         break;
1759     case DW_TAG_union_type:
1760         ctf_dprintf("union\n");
1761         ret = ctf_dwarf_create_sou(cup, die, idp, CTF_K_UNION,
1762         isroot);
1763         break;
1764     case DW_TAG_const_type:
1765         ctf_dprintf("const\n");
1766         ret = ctf_dwarf_create_reference(cup, die, idp, CTF_K_CONST,
1767         isroot);
1768         break;
1769     case DW_TAG_volatile_type:
1770         ctf_dprintf("volatile\n");
1771         ret = ctf_dwarf_create_reference(cup, die, idp, CTF_K_VOLATILE,
1772         isroot);
1773         break;
1774     case DW_TAG_restrict_type:
1775         ctf_dprintf("restrict\n");
1776         ret = ctf_dwarf_create_reference(cup, die, idp, CTF_K_RESTRICT,
1777         isroot);
1778         break;
1779     default:
1780         ctf_dprintf("ignoring tag type %x\n", tag);
1781         *idp = CTF_ERR;
1782         ret = 0;
1783         break;
1784     }
1785     ctf_dprintf("ctf_dwarf_convert_type tag specific handler returned %d\n",
1786     ret);
1788     return (ret);
1789 }

```

unchanged portion omitted

```

2630 /*
2631  * Note, we expect that if we're returning a ctf_file_t from one of the dies,
2632  * say in the single node case, it's been saved and the entry here has been set
2633  * to NULL, which ctf_close happily ignores.
2634  */
2635 static void
2636 ctf_dwarf_free_die(ctf_cu_t *cup)
2637 {
2638     ctf_dwfunc_t *cdf, *ndf;
2639     ctf_dwvar_t *cdv, *ndv;
2640     ctf_dwbitf_t *cdb, *ndb;
2641     ctf_dwmap_t *map;
2642     void *cookie;
2643     Dwarf_Error derr;
2645     ctf_dprintf("Beginning to free die: %p\n", cup);
2646     cup->cu_elf = NULL;
2647     ctf_dprintf("Trying to free name: %p\n", cup->cu_name);
2648     if (cup->cu_name != NULL)
2649         ctf_free(cup->cu_name, strlen(cup->cu_name) + 1);
2650     ctf_dprintf("Trying to free merge handle: %p\n", cup->cu_cmh);
2651     if (cup->cu_cmh != NULL) {
2652         ctf_merge_fini(cup->cu_cmh);
2653         cup->cu_cmh = NULL;
2654     }
2656     ctf_dprintf("Trying to free functions\n");
2657     for (cdf = ctf_list_next(&cup->cu_funcs); cdf != NULL; cdf = ndf) {
2658         ndf = ctf_list_next(cdf);
2659         ctf_free(cdf->cdf_name, strlen(cdf->cdf_name) + 1);

```

```

2660         if (cdf->cdf_fip.ctc_argc != 0) {
2661             ctf_free(cdf->cdf_argv,
2662             sizeof (ctf_id_t) * cdf->cdf_fip.ctc_argc);
2663         }
2664         ctf_free(cdf, sizeof (ctf_dwfunc_t));
2665     }
2667     ctf_dprintf("Trying to free variables\n");
2668     for (cdv = ctf_list_next(&cup->cu_vars); cdv != NULL; cdv = ndv) {
2669         ndv = ctf_list_next(cdv);
2670         ctf_free(cdv->cdv_name, strlen(cdv->cdv_name) + 1);
2671         ctf_free(cdv, sizeof (ctf_dwvar_t));
2672     }
2674     ctf_dprintf("Trying to free bitfields\n");
2675     for (cdb = ctf_list_next(&cup->cu_bitfields); cdb != NULL; cdb = ndb) {
2676         ndb = ctf_list_next(cdb);
2677         ctf_free(cdb, sizeof (ctf_dwbitf_t));
2678     }
2680     ctf_dprintf("Trying to clean up dwarf_t: %p\n", cup->cu_dwarf);
2681     if (cup->cu_dwarf != NULL)
2682         (void) dwarf_finish(cup->cu_dwarf, &derr);
2683     cup->cu_dwarf = NULL;
2684     ctf_close(cup->cu_ctfp);
2686     cookie = NULL;
2687     while ((map = avl_destroy_nodes(&cup->cu_map, &cookie)) != NULL) {
2688         ctf_free(map, sizeof (ctf_dwmap_t));
2689     }
2690     avl_destroy(&cup->cu_map);
2691     cup->cu_errbuf = NULL;
2692 }

```

unchanged portion omitted

```

2707 static int
2708 ctf_dwarf_count_dies(Dwarf_Debug dw, Dwarf_Error *derr, int *ndies,
2709     char *errbuf, size_t errlen)
2710 {
2711     int ret;
2712     Dwarf_Half vers;
2713     Dwarf_Unsigned nextthr;
2715     while ((ret = dwarf_next_cu_header(dw, NULL, &vers, NULL, NULL,
2716     &nextthr, derr)) != DW_DLV_NO_ENTRY) {
2717         if (ret != DW_DLV_OK) {
2718             (void) snprintf(errbuf, errlen,
2719             "file does not contain valid DWARF data: %s\n",
2720             dwarf_errmsg(*derr));
2721             return (ECTF_CONVBKERR);
2722         }
2724         if (vers != DWARF_VERSION_TWO) {
2725             (void) snprintf(errbuf, errlen,
2726             "unsupported DWARF version: %d\n", vers);
2727             return (ECTF_CONVBKERR);
2728         }
2729         *ndies = *ndies + 1;
2730     }
2730     if (*ndies == 0) {
2731         (void) snprintf(errbuf, errlen,
2732         "file does not contain valid DWARF data: %s\n",
2733         dwarf_errmsg(*derr));
2734         return (ECTF_CONVBKERR);
2735     }

```



```

2732     return (0);
2733 }

2735 static int
2736 ctf_dwarf_init_die(int fd, Elf *elf, ctf_cu_t *cup, int ndie, char *errbuf,
2737                  size_t errlen)
2738 {
2739     int ret;
2740     Dwarf_Unsigned hdrlen, abboff, nexthdr;
2741     Dwarf_Half addrsz;
2742     Dwarf_Unsigned offset = 0;
2743     Dwarf_Error derr;

2745     while ((ret = dwarf_next_cu_header(cup->cu_dwarf, &hdrlen, NULL,
2746                                       &abboff, &addrsz, &nexthdr, &derr)) != DW_DLV_NO_ENTRY) {
2747         char *name;
2748         Dwarf_Die cu, child;

2750         /* Based on the counting above, we should be good to go */
2751         VERIFY(ret == DW_DLV_OK);
2752         if (ndie > 0) {
2753             ndie--;
2754             offset = nexthdr;
2755             continue;
2756         }

2758         /*
2759          * Compilers are apparently inconsistent. Some emit no DWARF for
2760          * empty files and others emit empty compilation unit.
2761          */
2762         cup->cu_voidtid = CTF_ERR;
2763         cup->cu_longtid = CTF_ERR;
2764         cup->cu_elf = elf;
2765         cup->cu_maxoff = nexthdr - 1;
2766         cup->cu_ctfp = ctf_fdcreate(fd, &ret);
2767         if (cup->cu_ctfp == NULL)
2768             if (cup->cu_ctfp == NULL) {
2769                 ctf_free(cup, sizeof (ctf_cu_t));
2770                 return (ret);
2771             }

2772         avl_create(&cup->cu_map, ctf_dwmap_comp, sizeof (ctf_dwmap_t),
2773                 offsetof(ctf_dwmap_t, cdm_avl));
2774         cup->cu_errbuf = errbuf;
2775         cup->cu_errlen = errlen;
2776         bzero(&cup->cu_vars, sizeof (ctf_list_t));
2777         bzero(&cup->cu_funcs, sizeof (ctf_list_t));
2778         bzero(&cup->cu_bitfields, sizeof (ctf_list_t));

2779         if ((ret = ctf_dwarf_die_elfenc(elf, cup, errbuf,
2780                                       errlen)) != 0) {
2781             avl_destroy(&cup->cu_map);
2782             ctf_free(cup, sizeof (ctf_cu_t));
2783             return (ret);
2784         }

2785         if ((ret = ctf_dwarf_sib(cup, NULL, &cu)) != 0)
2786             if ((ret = ctf_dwarf_sib(cup, NULL, &cu)) != 0) {
2787                 avl_destroy(&cup->cu_map);
2788                 ctf_free(cup, sizeof (ctf_cu_t));
2789                 return (ret);
2790             }

2791         if (cu == NULL) {

```

```

2786         (void) snprintf(errbuf, errlen,
2787                        "file does not contain DWARF data");
2788         return (ECTF_CONVNODEBUG);
2789         "file does not contain DWARF data\n");
2790     avl_destroy(&cup->cu_map);
2791     ctf_free(cup, sizeof (ctf_cu_t));
2792     return (ECTF_CONVBKERR);
2793 }

2794 if ((ret = ctf_dwarf_child(cup, cu, &child)) != 0)
2795     if ((ret = ctf_dwarf_child(cup, cu, &child)) != 0) {
2796         avl_destroy(&cup->cu_map);
2797         ctf_free(cup, sizeof (ctf_cu_t));
2798         return (ret);
2799     }

2800     if (child == NULL) {
2801         (void) snprintf(errbuf, errlen,
2802                        "file does not contain DWARF data");
2803         return (ECTF_CONVNODEBUG);
2804         "file does not contain DWARF data\n");
2805     avl_destroy(&cup->cu_map);
2806     ctf_free(cup, sizeof (ctf_cu_t));
2807     return (ECTF_CONVBKERR);
2808 }

2809     cup->cu_cuoff = offset;
2810     cup->cu_cu = child;

2811     if ((cup->cu_cmh = ctf_merge_init(fd, &ret)) == NULL)
2812         if ((cup->cu_cmh = ctf_merge_init(fd, &ret)) == NULL) {
2813             avl_destroy(&cup->cu_map);
2814             ctf_free(cup, sizeof (ctf_cu_t));
2815             return (ret);
2816         }

2817     if (ctf_dwarf_string(cup, cu, DW_AT_name, &name) == 0) {
2818         size_t len = strlen(name) + 1;
2819         char *b = basename(name);
2820         cup->cu_name = strdup(b);
2821         ctf_free(name, len);
2822     }
2823     break;
2824 }

2825     return (0);
2826 }

2827 /*
2828  * This is our only recourse to identify a C source file that is missing debug
2829  * info: it will be mentioned as an STT_FILE, but not have a compile unit entry.
2830  * (A traditional ctfmerge works on individual files, so can identify missing
2831  * DWARF more directly, via ctf_has_c_source() on the .o file.)
2832  *
2833  * As we operate on basenames, this can of course miss some cases, but it's
2834  * better than not checking at all.
2835  *
2836  * We explicitly whitelist some CRT components. Failing that, there's always
2837  * the -m option.
2838  */
2839 static boolean_t
2840 c_source_has_debug(const char *file, ctf_cu_t *cus, size_t nr_cus)
2841 {
2842     const char *basename = strrchr(file, '/');
2843     if (basename == NULL)

```

```

2836     basename = file;
2837     else
2838         basename++;
2840     if (strcmp(basename, "common-crt.c") == 0 ||
2841         strcmp(basename, "gmon.c") == 0 ||
2842         strcmp(basename, "dlink_init.c") == 0 ||
2843         strcmp(basename, "dlink_common.c") == 0 ||
2844         strncmp(basename, "crt", strlen("crt")) == 0 ||
2845         strncmp(basename, "values-", strlen("values-")) == 0)
2846         return (B_TRUE);
2848     for (size_t i = 0; i < nr_cus; i++) {
2849         if (strcmp(basename, cus[i].cu_name) == 0)
2850             return (B_TRUE);
2851     }
2853     return (B_FALSE);
2854 }
2856 static int
2857 ctf_dwarf_check_missing(ctf_cu_t *cus, size_t nr_cus, Elf *elf,
2858 ctf_conv_status_t
2859 ctf_dwarf_convert(int fd, Elf *elf, uint_t nthrs, int *errp, ctf_file_t **fpp,
2858 char *errmsg, size_t errlen)
2859 {
2860     Elf_Scn *scn, *strscn;
2861     Elf_Data *data, *strdata;
2862     GElf_Shdr shdr;
2863     ulong_t i;
2865     scn = NULL;
2866     while ((scn = elf_nextscn(elf, scn)) != NULL) {
2867         if (gelf_getshdr(scn, &shdr) == NULL) {
2868             (void) snprintf(errmsg, errlen,
2869                 "failed to get section header: %s\n",
2870                 elf_errmsg(elf_errno()));
2871             return (EINVAL);
2872         }
2874         if (shdr.sh_type == SHT_SYMTAB)
2875             break;
2876     }
2878     if (scn == NULL)
2879         return (0);
2881     if ((strscn = elf_getscn(elf, shdr.sh_link)) == NULL) {
2882         (void) snprintf(errmsg, errlen,
2883             "failed to get str section: %s\n",
2884             elf_errmsg(elf_errno()));
2885         return (EINVAL);
2886     }
2888     if ((data = elf_getdata(scn, NULL)) == NULL) {
2889         (void) snprintf(errmsg, errlen, "failed to read section: %s\n",
2890             elf_errmsg(elf_errno()));
2891         return (EINVAL);
2892     }
2894     if ((strdata = elf_getdata(strscn, NULL)) == NULL) {
2895         (void) snprintf(errmsg, errlen,
2896             "failed to read string table: %s\n",
2897             elf_errmsg(elf_errno()));
2898         return (EINVAL);
2899     }

```

```

2901     for (i = 0; i < shdr.sh_size / shdr.sh_entsize; i++) {
2902         GElf_Sym sym;
2903         const char *file;
2904         size_t len;
2906         if (gelf_getsym(data, i, &sym) == NULL) {
2907             (void) snprintf(errmsg, errlen,
2908                 "failed to read sym %lu: %s\n",
2909                 i, elf_errmsg(elf_errno()));
2910             return (EINVAL);
2911         }
2913         if (GELF_ST_TYPE(sym.st_info) != STT_FILE)
2914             continue;
2916         file = (const char *)((uintptr_t)strdata->d_buf + sym.st_name);
2917         len = strlen(file);
2918         if (len < 2 || strcmp(".c", &file[len - 2], 2) != 0)
2919             continue;
2921         if (!source_has_debug(file, cus, nr_cus)) {
2922             (void) snprintf(errmsg, errlen,
2923                 "file %s is missing debug info\n", file);
2924             return (ECTF_CONVNODEBUG);
2925         }
2926     }
2928     return (0);
2929 }
2931 int
2932 ctf_dwarf_convert(int fd, Elf *elf, uint_t nthrs, uint_t flags,
2933 ctf_file_t **fpp, char *errbuf, size_t errlen)
2934 {
2935     int err, ret, ndies, i;
2936     Dwarf_Debug dw;
2937     Dwarf_Error derr;
2938     ctf_cu_t *cdies = NULL, *cup;
2939     workq_t *wqp = NULL;
2941     if (errp == NULL)
2942         errp = &err;
2943     *errp = 0;
2944     *fpp = NULL;
2946     ret = dwarf_elf_init(elf, DW_DLC_READ, NULL, NULL, &dw, &derr);
2947     if (ret != DW_DLV_OK) {
2948         /*
2949          * We may want to expect DWARF data here and fail conversion if
2950          * it's missing. In this case, if we actually have some amount
2951          * of DWARF, but no section, for now, just go ahead and create
2952          * an empty CTF file.
2953          */
2954         if (ret == DW_DLV_NO_ENTRY ||
2955             dwarf_errno(derr) == DW_DLE_DEBUG_INFO_NULL) {
2956             (void) snprintf(errbuf, errlen,
2957                 "file does not contain DWARF data\n");
2958             return (ECTF_CONVNODEBUG);
2959             *fpp = ctf_create(errp);
2960             return (*fpp != NULL ? CTF_CONV_SUCCESS :
2961                 CTF_CONV_ERROR);
2962         }
2963     }
2965     (void) snprintf(errbuf, errlen,
2966         "dwarf_elf_init() failed: %s\n", dwarf_errmsg(derr));

```

```

2954     return (ECTF_CONVBKERR);
2955     (void) snprintf(errmsg, errlen,
2956     "failed to initialize DWARF: %s\n",
2957     dwarf_errmsg(derr));
2958     *errp = ECTF_CONVBKERR;
2959     return (CTF_CONV_ERROR);
2960 }
2961
2962 /*
2963  * Iterate over all of the compilation units and create a ctf_cu_t for
2964  * each of them. This is used to determine if we have zero, one, or
2965  * multiple dies to convert. If we have zero, that's an error. If
2966  * there's only one die, that's the simple case. No merge needed and
2967  * only a single Dwarf_Debug as well.
2968  */
2969 ndies = 0;
2970 err = ctf_dwarf_count_dies(dw, &derr, &ndies, errbuf, errlen);
2971
2972 ctf_dprintf("found %d DWARF CUs\n", ndies);
2973
2974 if (ndies == 0) {
2975     (void) snprintf(errbuf, errlen,
2976     "file does not contain DWARF data\n");
2977     return (ECTF_CONVNODEBUG);
2978 }
2979 ret = ctf_dwarf_count_dies(dw, &derr, &ndies, errmsg, errlen);
2980 if (ret != 0) {
2981     *errp = ret;
2982     goto out;
2983 }
2984
2985 (void) dwarf_finish(dw, &derr);
2986 cdies = ctf_alloc(sizeof (ctf_cu_t) * ndies);
2987 if (cdies == NULL) {
2988     return (ENOMEM);
2989     *errp = ENOMEM;
2990     return (CTF_CONV_ERROR);
2991 }
2992
2993 bzero(cdies, sizeof (ctf_cu_t) * ndies);
2994
2995 for (i = 0; i < ndies; i++) {
2996     cup = &cdies[i];
2997     ret = dwarf_elf_init(elf, DW_DLC_READ, NULL, NULL,
2998     &cup->cu_dwarf, &derr);
2999     if (ret != 0) {
3000         ctf_free(cdies, sizeof (ctf_cu_t) * ndies);
3001         (void) snprintf(errbuf, errlen,
3002         (void) snprintf(errmsg, errlen,
3003         "failed to initialize DWARF: %s\n",
3004         dwarf_errmsg(derr));
3005         return (ECTF_CONVBKERR);
3006         *errp = ECTF_CONVBKERR;
3007         return (CTF_CONV_ERROR);
3008     }
3009 }
3010
3011 err = ctf_dwarf_init_die(fd, elf, cup, i, errbuf, errlen);
3012 if (err != 0)
3013     ret = ctf_dwarf_init_die(fd, elf, &cdies[i], i, errmsg, errlen);
3014 if (ret != 0) {
3015     *errp = ret;
3016     goto out;
3017 }
3018
3019 cup->cu_doweaks = ndies > 1 ? B_FALSE : B_TRUE;
3020 }

```

```

3002     if (!(flags & CTF_ALLOW_MISSING_DEBUG) &&
3003     (err = ctf_dwarf_check_missing(cdies, ndies,
3004     elf, errbuf, errlen)) != 0)
3005         goto out;
3006     ctf_dprintf("found %d DWARF CUs\n", ndies);
3007
3008     /*
3009     * If we only have one compilation unit, there's no reason to use
3010     * multiple threads, even if the user requested them. After all, they
3011     * just gave us an upper bound.
3012     */
3013     if (ndies == 1)
3014         nthrs = 1;
3015
3016     if (workq_init(&wqp, nthrs) == -1) {
3017         err = errno;
3018         *errp = errno;
3019         goto out;
3020     }
3021
3022     for (i = 0; i < ndies; i++) {
3023         cup = &cdies[i];
3024         ctf_dprintf("adding cu %s: %p, %x %x\n", cup->cu_name,
3025         cup->cu_cu, cup->cu_cuoff, cup->cu_maxoff);
3026         if (workq_add(wqp, cup) == -1) {
3027             err = errno;
3028             *errp = errno;
3029             goto out;
3030         }
3031     }
3032
3033     ret = workq_work(wqp, ctf_dwarf_convert_one, NULL, &err);
3034     ret = workq_work(wqp, ctf_dwarf_convert_one, NULL, errp);
3035     if (ret == WORKQ_ERROR) {
3036         err = errno;
3037         *errp = errno;
3038         goto out;
3039     } else if (ret == WORKQ_UERROR) {
3040         ctf_dprintf("internal convert failed: %s\n",
3041         ctf_errmsg(err));
3042         ctf_errmsg(*errp);
3043         goto out;
3044     }
3045
3046     ctf_dprintf("Determining next phase: have %d CUs\n", ndies);
3047     if (ndies != 1) {
3048         ctf_merge_t *cmp;
3049
3050         cmp = ctf_merge_init(fd, &err);
3051         if (cmp == NULL)
3052             cmp = ctf_merge_init(fd, &ret);
3053         if (cmp == NULL) {
3054             *errp = ret;
3055             goto out;
3056         }
3057
3058         ctf_dprintf("setting threads\n");
3059         if ((err = ctf_merge_set_nthreads(cmp, nthrs)) != 0) {
3060             if ((ret = ctf_merge_set_nthreads(cmp, nthrs)) != 0) {
3061                 ctf_merge_fini(cmp);
3062                 *errp = ret;
3063                 goto out;
3064             }
3065         }
3066
3067         for (i = 0; i < ndies; i++) {
3068             cup = &cdies[i];

```

```
3056         if ((err = ctf_merge_add(cmp, cup->cu_ctfp)) != 0) {
3072             ctf_dprintf("adding cu %s (%p)\n", cup->cu_name,
3073                 cup->cu_ctfp);
3074             if ((ret = ctf_merge_add(cmp, cup->cu_ctfp)) != 0) {
3057                 ctf_merge_fini(cmp);
3076                 *errp = ret;
3058             }
3059         }
3060     }

3062     ctf_dprintf("performing merge\n");
3063     err = ctf_merge_merge(cmp, fpp);
3064     if (err != 0) {
3082         ret = ctf_merge_merge(cmp, fpp);
3083         if (ret != 0) {
3065             ctf_dprintf("failed merge!\n");
3066             *fpp = NULL;
3067             ctf_merge_fini(cmp);
3087             *errp = ret;
3068             goto out;
3069         }
3070         ctf_merge_fini(cmp);
3071         err = 0;
3091         *errp = 0;
3072         ctf_dprintf("successfully converted!\n");
3073     } else {
3074         err = 0;
3094         *errp = 0;
3075         *fpp = cdies->cu_ctfp;
3076         cdies->cu_ctfp = NULL;
3077         ctf_dprintf("successfully converted!\n");
3078     }

3080 out:
3081     workq_fini(wqp);
3082     ctf_dwarf_free_dies(cdies, ndies);
3083     return (err);
3003     return (*fpp != NULL ? CTF_CONV_SUCCESS : CTF_CONV_ERROR);
3084 }

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libctf/common/libctf.h

1

```
*****
3843 Thu Apr 25 02:20:44 2019
new/usr/src/lib/libctf/common/libctf.h
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23 * Copyright 2001-2003 Sun Microsystems, Inc. All rights reserved.
24 * Use is subject to license terms.
25 */
26 /*
27 * Copyright (c) 2019, Joyent, Inc.
27 * Copyright (c) 2015, Joyent, Inc.
28 */

30 /*
31 * This header file defines the interfaces available from the CTF debugger
32 * library, libctf. This library provides functions that a debugger can
33 * use to operate on data in the Compact ANSI-C Type Format (CTF). This
34 * is NOT a public interface, although it may eventually become one in
35 * the fullness of time after we gain more experience with the interfaces.
36 *
37 * In the meantime, be aware that any program linked with libctf in this
38 * release of illumos is almost guaranteed to break in the next release.
39 *
40 * In short, do not user this header file or libctf for any purpose.
41 */

43 #ifndef _LIBCTF_H
44 #define _LIBCTF_H

46 #include <sys/ctf_api.h>
47 #include <libelf.h>

49 #ifdef __cplusplus
50 extern "C" {
51 #endif
```

new/usr/src/lib/libctf/common/libctf.h

2

```
53 /*
54  * This flag can be used to enable debug messages.
55  */
56 extern int _libctf_debug;

58 typedef enum ctf_diff_flag {
59     CTF_DIFF_F_IGNORE_INTNAMES = 0x01
60 } ctf_diff_flag_t;

62 typedef struct ctf_diff ctf_diff_t;
63 typedef void (*ctf_diff_type_f)(ctf_file_t *, ctf_id_t, boolean_t, ctf_file_t *,
64     ctf_id_t, void *);
65 typedef void (*ctf_diff_func_f)(ctf_file_t *, ulong_t, boolean_t, ctf_file_t *,
66     ulong_t, void *);
67 typedef void (*ctf_diff_obj_f)(ctf_file_t *, ulong_t, ctf_id_t, boolean_t,
68     ctf_file_t *, ulong_t, ctf_id_t, void *);

70 extern int ctf_diff_init(ctf_file_t *, ctf_file_t *, ctf_diff_t **);
71 extern uint_t ctf_diff_getflags(ctf_diff_t *);
72 extern int ctf_diff_setflags(ctf_diff_t *, uint_t);
73 extern int ctf_diff_types(ctf_diff_t *, ctf_diff_type_f, void *);
74 extern int ctf_diff_functions(ctf_diff_t *, ctf_diff_func_f, void *);
75 extern int ctf_diff_objects(ctf_diff_t *, ctf_diff_obj_f, void *);
76 extern void ctf_diff_fini(ctf_diff_t *);

78 /*
79  * Normally, we return a failure if we find a C-derived compilation unit that
80  * lacks DWARF or CTF (as required). This flag over-rides this check.
81  */
82 #define CTF_ALLOW_MISSING_DEBUG 0x01

77 #define CTF_CONVERT_F_IGNNONC 0x01
84 extern ctf_file_t *ctf_fdconvert(int, const char *, uint_t, uint_t, int *,
85     char *, size_t);

87 typedef enum ctf_hsc_ret {
88     CHR_ERROR = -1,
89     CHR_NO_C_SOURCE = 0,
90     CHR_HAS_C_SOURCE = 1
91 } ctf_hsc_ret_t;

93 extern ctf_hsc_ret_t ctf_has_c_source(Elf *, char *, size_t);

95 typedef struct ctf_merge_handle ctf_merge_t;
96 extern ctf_merge_t *ctf_merge_init(int, int *);
97 extern int ctf_merge_add(ctf_merge_t *, ctf_file_t *);
98 extern int ctf_merge_set_nthreads(ctf_merge_t *, const uint_t);
99 extern int ctf_merge_label(ctf_merge_t *, const char *);
100 extern int ctf_merge_uniquify(ctf_merge_t *, ctf_file_t *, const char *);
101 extern int ctf_merge_merge(ctf_merge_t *, ctf_file_t **);
102 extern int ctf_merge_dedup(ctf_merge_t *, ctf_file_t **);
103 extern void ctf_merge_fini(ctf_merge_t *);

105 #define CTF_ELFWRITE_F_COMPRESS 0x1
106 extern int ctf_elffwrite(ctf_file_t *, int, int, int);
107 extern int ctf_elfwrite(ctf_file_t *, const char *, const char *, int);

109 #ifdef __cplusplus
110 }
    unchanged_portion_omitted

```

new/usr/src/lib/libctf/common/libctf_impl.h

1

1502 Thu Apr 25 02:20:44 2019

new/usr/src/lib/libctf/common/libctf_impl.h

10816 ctf_dwarf_convert_type() relies on uninitialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 at
9  * http://www.illumos.org/license/CDDL.
10 */
12 /*
13  * Copyright 2019 Joyent, Inc.
14 */
16 #ifndef _LIBCTF_IMPL_H
17 #define _LIBCTF_IMPL_H
19 /*
20  * Portions of libctf implementations that are only suitable for CTF's userland
21  * library, eg. converting and merging related routines.
22 */
24 #include <libelf.h>
25 #include <libctf.h>
26 #include <ctf_impl.h>
28 #ifdef __cplusplus
29 extern "C" {
30 #endif
32 typedef int (*ctf_convert_f)(int, Elf *, uint_t, uint_t,
32 typedef enum ctf_conv_status {
33     CTF_CONV_SUCCESS      = 0,
34     CTF_CONV_ERROR       = 1,
35     CTF_CONV_NOTSUP      = 2
36 } ctf_conv_status_t;
38 typedef ctf_conv_status_t (*ctf_convert_f)(int, Elf *, uint_t, int *,
33     ctf_file_t **, char *, size_t);
34 extern int ctf_dwarf_convert(int, Elf *, uint_t, uint_t,
40 extern ctf_conv_status_t ctf_dwarf_convert(int, Elf *, uint_t, int *,
35     ctf_file_t **, char *, size_t);
37 /*
38  * Symbol walking
39 */
40 typedef int (*ctf_syntab_f)(const Elf64_Sym *, ulong_t, const char *,
41     const char *, boolean_t, void *);
42 extern int ctf_syntab_iter(ctf_file_t *, ctf_syntab_f, void *);
44 /*
45  * zlib compression routines
```

new/usr/src/lib/libctf/common/libctf_impl.h

2

```
46 */
47 extern int ctf_compress(ctf_file_t *fp, void **, size_t *, size_t *);
49 extern int ctf_diff_self(ctf_diff_t *, ctf_diff_type_f, void *);
51 /*
52  * Internal debugging aids
53 */
54 extern void ctf_phase_dump(ctf_file_t *, const char *, const char *);
55 extern void ctf_phase_bump(void);
57 #ifdef __cplusplus
58 }
    unchanged_portion_omitted
```

```

*****
3257 Thu Apr 25 02:20:44 2019
new/usr/src/lib/libctf/common/mapfile-vers
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
23 #
24 #
25 #
26 # Copyright 2019 Joyent, Inc.
27 # Copyright 2018 Joyent, Inc.
28 #
29 #
30 # MAPFILE HEADER START
31 #
32 # WARNING: STOP NOW. DO NOT MODIFY THIS FILE.
33 # Object versioning must comply with the rules detailed in
34 #
35 #     usr/src/lib/README.mapfiles
36 #
37 # You should not be making modifications here until you've read the most current
38 # copy of that file. If you need help, contact a gatekeeper for guidance.
39 #
40 # MAPFILE HEADER END
41 #
42 #
43 $mapfile_version 2
44 #
45 # There really should be only one SUNWprivate version.
46 # Don't add any more. Add new private symbols to SUNWprivate_1.2
47 #
48 SYMBOL_VERSION SUNWprivate_1.2 {
49     global:
50         ctf_add_array;
51         ctf_add_const;
52         ctf_add_enum;

```

```

53     ctf_add_enumerator;
54     ctf_add_float;
55     ctf_add_forward;
56     ctf_add_funcptr;
57     ctf_add_function;
58     ctf_add_integer;
59     ctf_add_label;
60     ctf_add_member;
61     ctf_add_object;
62     ctf_add_pointer;
63     ctf_add_restrict;
64     ctf_add_struct;
65     ctf_add_type;
66     ctf_add_typedef;
67     ctf_add_union;
68     ctf_add_volatile;
69     ctf_create;
70     ctf_dataptr;
71     ctf_delete_type;
72     ctf_diff_fini;
73     ctf_diff_functions;
74     ctf_diff_getflags;
75     ctf_diff_init;
76     ctf_diff_objects;
77     ctf_diff_setflags;
78     ctf_diff_types;
79     ctf_discard;
80     ctf_dup;
81     ctf_elffdwrite;
82     ctf_elfwrite;
83     ctf_enum_value;
84     ctf_fdconvert;
85     ctf_flags;
86     ctf_func_args_by_id;
87     ctf_func_info_by_id;
88     ctf_function_iter;
89     ctf_has_c_source;
90     ctf_kind_name;
91     ctf_label_info;
92     ctf_label_iter;
93     ctf_label_topmost;
94     ctf_max_id;
95     ctf_member_info;
96     ctf_merge_add;
97     ctf_merge_dedup;
98     ctf_merge_fini;
99     ctf_merge_init;
100    ctf_merge_label;
101    ctf_merge_merge;
102    ctf_merge_set_nthreads;
103    ctf_merge_uniquify;
104    ctf_nr_syms;
105    ctf_object_iter;
106    ctf_parent_file;
107    ctf_parent_label;
108    ctf_parent_name;
109    ctf_set_array;
110    ctf_set_root;
111    ctf_set_size;
112    ctf_string_iter;
113    ctf_symbol_name;
114    ctf_type_align;
115    ctf_type_cmp;
116    ctf_type_cname;
117    ctf_type_compat;
118    ctf_type_pointer;

```

new/usr/src/lib/libctf/common/mapfile-vers

3

```
119         ctf_update;  
120         ctf_write;  
121 } SUNWprivate_1.1;  
_____unchanged_portion_omitted_____
```


new/usr/src/lib/libfru/Makefile.targ

1

```
*****
1856 Thu Apr 25 02:20:45 2019
new/usr/src/lib/libfru/Makefile.targ
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
```

```
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License, Version 1.0 only
6 # (the "License"). You may not use this file except in compliance
7 # with the License.
8 #
9 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 # or http://www.opensolaris.org/os/licensing.
11 # See the License for the specific language governing permissions
12 # and limitations under the License.
13 #
14 # When distributing Covered Code, include this CDDL HEADER in each
15 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 # If applicable, add the following below this CDDL HEADER, with the
17 # fields enclosed by brackets "[]" replaced with your own identifying
18 # information: Portions Copyright [yyyy] [name of copyright owner]
19 #
20 # CDDL HEADER END
21 #
22 #
23 # Copyright (c) 2000 by Sun Microsystems, Inc.
24 # All rights reserved.
25 #
26 # Copyright (c) 2019, Joyent, Inc.
26 #pragma ident "%Z%M% %I% %E% SMI"
27 #
28 # lib/libfru/Makefile.targ
29 #

29 include ../../Makefile.targ

31 # build rules for pics. (access sub directories)
32 pics/%.o objs/%.o: ../libfru/%.cc
33 $(COMPILE.cc) -o $@ $<
34 $(POST_PROCESS_CC_O)
36 $(POST_PROCESS_O)

36 pics/%.o objs/%.o: ../libfrureg/%.cc
37 $(COMPILE.cc) -o $@ $<
38 $(POST_PROCESS_CC_O)
40 $(POST_PROCESS_O)

40 pics/%.o objs/%.o: ../libgenutil/%.cc
41 $(COMPILE.cc) -o $@ $<
42 $(POST_PROCESS_CC_O)
44 $(POST_PROCESS_O)

44 # some special rules for the yacc and lex parser. (with sub directories)
45 pics/y.tab.cc pics/y.tab.h: ../libfru/nameSyntaxYacc.y
46 $(YACC.y) -d ../libfru/nameSyntaxYacc.y
47 $(SED) -f ../libfru/yy-sed y.tab.c > pics/y.tab.cc
```

new/usr/src/lib/libfru/Makefile.targ

2

```
48 $(SED) -f ../libfru/yy-sed y.tab.h > pics/y.tab.h
49 $(RM) y.tab.c y.tab.h

51 pics/nameSyntaxYacc.o: pics/y.tab.cc
52 $(COMPILE.cc) -I../libfru -Ipics -o $@ pics/y.tab.cc

54 pics/nameSyntaxLex.o: ../libfru/nameSyntaxLex.l pics/y.tab.h
55 $(LEX.l) ../libfru/nameSyntaxLex.l | \
56 $(SED) -f ../libfru/yy-lsed > pics/lex.fru.cc
57 $(COMPILE.cc) -I../libfru -Ipics -o $@ pics/lex.fru.cc
```

new/usr/src/lib/libm/Makefile.targ

1

```
*****
1158 Thu Apr 25 02:20:45 2019
new/usr/src/lib/libm/Makefile.targ
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
```

```
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 at
9 # http://www.illumos.org/license/CDDL.
10 #
```

```
12 #
13 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
14 #
15 # Copyright (c) 2019, Joyent, Inc.
16 #
```

```
18 pics/%.o: $(LIBMDIR)/$(TARGETMACH)/src/%.$(ASSUFFIX)
19     $(COMPILE.s) -o $@ $<
20     $(POST_PROCESS_S_O)
18     $(POST_PROCESS_O)
```

```
22 pics/%.o: $(SRCDIR)/C/%.c $(LM_IL)
23     $(COMPILE.c) -o $@ $<
24     $(POST_PROCESS_O)
```

```
26 pics/%.o: $(SRCDIR)/$(LDBLDIR)/%.c $(LM_IL)
27     $(COMPILE.c) -o $@ $<
28     $(POST_PROCESS_O)
```

```
30 pics/%.o: $(SRCDIR)/R/%.c $(LM_IL)
31     $(COMPILE.c) -o $@ $<
32     $(POST_PROCESS_O)
```

```
34 pics/%.o: $(SRCDIR)/complex/%.c $(LM_IL)
35     $(COMPILE.c) -o $@ $<
36     $(POST_PROCESS_O)
```

```
38 pics/%.o: $(SRCDIR)/m9x/%.c $(LM_IL) $(m9x_IL)
39     $(COMPILE.c) $(m9x_IL) -o $@ $<
40     $(POST_PROCESS_O)
```

```
42 $(ROOTLIBDIR): $(ROOTFS_LIBDIR)
43     $(INS.dir)
```

```
45 $(ROOTLIBDIR64): $(ROOTFS_LIBDIR64)
46     $(INS.dir)
```

```
48 include $(SRC)/lib/Makefile.targ
```

```

*****
5598 Thu Apr 25 02:20:45 2019
new/usr/src/lib/libmvec/Makefile.com
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
14 #
15 # Copyright (c) 2019, Joyent, Inc.
16 #
17 #
18 LIBMDIR      = $(SRC)/lib/libm
19 #
20 mvecOBSJS    = \
21     _vTBL_atan1.o \
22     _vTBL_atan2.o \
23     _vTBL_rsqrt.o \
24     _vTBL_sincos.o \
25     _vTBL_sincos2.o \
26     _vTBL_sqrtf.o \
27     _vatan.o \
28     _vatan2.o \
29     _vatan2f.o \
30     _vatanf.o \
31     _vc_abs.o \
32     _vc_exp.o \
33     _vc_log.o \
34     _vc_pow.o \
35     _vcos.o \
36     _vcosbig.o \
37     _vcosbigf.o \
38     _vcosf.o \
39     _vexp.o \
40     _vexpf.o \
41     _vhypot.o \
42     _vhypotf.o \
43     _vlog.o \
44     _vlogf.o \
45     _vpow.o \
46     _vpowf.o \
47     _vrem_pio2m.o \
48     _vrhypot.o \
49     _vrhypotf.o \
50     _vrsqrt.o \
51     _vrsqrtf.o \
52     _vsin.o \
53     _vsinbig.o \

```

```

54     _vsinbigf.o \
55     _vsincos.o \
56     _vsincosbig.o \
57     _vsincosbigf.o \
58     _vsincosf.o \
59     _vsinf.o \
60     _vsqrt.o \
61     _vsqrtf.o \
62     _vz_abs.o \
63     _vz_exp.o \
64     _vz_log.o \
65     _vz_pow.o \
66     vatan2.o \
67     vatan2f.o \
68     vatan.o \
69     vatanf.o \
70     vc_abs.o \
71     vc_exp.o \
72     vc_log.o \
73     vc_pow.o \
74     vcos.o \
75     vcosf.o \
76     vexp.o \
77     vexpf.o \
78     vhypot.o \
79     vhypotf.o \
80     vlog.o \
81     vlogf.o \
82     vpow.o \
83     vpowf.o \
84     vrhypot.o \
85     vrhypotf.o \
86     vrsqrt.o \
87     vrsqrtf.o \
88     vsin.o \
89     vsincos.o \
90     vsincosf.o \
91     vsinf.o \
92     vsqrt.o \
93     vsqrtf.o \
94     vz_abs.o \
95     vz_exp.o \
96     vz_log.o \
97     vz_pow.o
98 #
99 mvecvisCOBJS = \
100     _vTBL_atan1.o \
101     _vTBL_atan2.o \
102     _vTBL_rsqrt.o \
103     _vTBL_sincos.o \
104     _vTBL_sincos2.o \
105     _vTBL_sqrtf.o \
106     _vcosbig.o \
107     _vcosbigf.o \
108     _vrem_pio2m.o \
109     _vsinbig.o \
110     _vsinbigf.o \
111     _vsincosbig.o \
112     _vsincosbigf.o
113 #
114 mvecvisSOBJS = \
115     _vatan.o \
116     _vatan2.o \
117     _vatan2f.o \
118     _vatanf.o \
119     _vcos.o \

```

```

120      __vcosf.o \
121      __vexp.o \
122      __vexpf.o \
123      __vhypot.o \
124      __vhypotf.o \
125      __vlog.o \
126      __vlogf.o \
127      __vpow.o \
128      __vpowf.o \
129      __vrhypot.o \
130      __vrhypotf.o \
131      __vrsqrt.o \
132      __vrsqrtf.o \
133      __vsin.o \
134      __vsincos.o \
135      __vsincosf.o \
136      __vsinf.o \
137      __vsqrt.o \
138      __vsqrtf.o

140 mvecvis2COBJS = \
141     __vTBL_sincos.o \
142     __vTBL_sincos2.o \
143     __vTBL_sqrtf.o \
144     __vcosbig.o \
145     __vcosbig_ultra3.o \
146     __vrem_pio2m.o \
147     __vsinbig.o \
148     __vsinbig_ultra3.o

150 mvecvis2SOBJS = \
151     __vcos_ultra3.o \
152     __vlog_ultra3.o \
153     __vsin_ultra3.o \
154     __vsqrtf_ultra3.o

156 include $(SRC)/lib/Makefile.lib
157 include $(SRC)/lib/Makefile.rootfs
158 include $(LIBMDIR)/Makefile.libm.com

160 LIBS = $(DYNLIB)
161 SRCDIR = ../common/
162 DYNFLAGS += $(ZIGNORE)

164 LINTERROFF = -erroff=E_FP_DIVISION_BY_ZERO
165 LINTERROFF += -erroff=E_FP_INVALID
166 LINTERROFF += -erroff=E_BAD_PTR_CAST_ALIGN
167 LINTERROFF += -erroff=E_ASSIGNMENT_CAUSE_LOSS_PREC
168 LINTERROFF += -erroff=E_FUNC_SET_NOT_USED

170 LINTFLAGS += $(LINTERROFF)
171 LINTFLAGS64 += $(LINTERROFF)
172 LINTFLAGS64 += -errchk=longptr64

174 CLAGS += $(LINTERROFF)
175 CFLAGS64 += $(LINTERROFF)

177 FLTRPATH_sparc = $$ORIGIN/cpu/$$ISALIST/libmvec_isa.so.1
178 FLTRPATH_sparcv9 = $$ORIGIN/../cpu/$$ISALIST/sparcv9/libmvec_isa.so.1
179 FLTRPATH_i386 = $$ORIGIN/libmvec/$$HWCAP
180 FLTRPATH = $(FLTRPATH_$(TARGET_ARCH))

182 sparc_CFLAGS += -_cc=-W0,-xintrinsic
183 sparcv9_CFLAGS += -_cc=-W0,-xintrinsic

185 SRCS_mvec_i386 = ../common/__vsqrtf.c

```

```

187 SRCS_mvec_sparc = $(SRCS_mvec_i386)

189 SRCS_mvec_sparcv9 = $(SRCS_mvec_i386)

192 SRCS_mvec = \
193     $(SRCS_mvec_$(TARGETMACH)) \
194     ../common/__vTBL_atan1.c \
195     ../common/__vTBL_atan2.c \
196     ../common/__vTBL_rsqr.c \
197     ../common/__vTBL_sincos.c \
198     ../common/__vTBL_sincos2.c \
199     ../common/__vTBL_sqrtf.c \
200     ../common/__vatan.c \
201     ../common/__vatan2.c \
202     ../common/__vatan2f.c \
203     ../common/__vatanf.c \
204     ../common/__vc_abs.c \
205     ../common/__vc_exp.c \
206     ../common/__vc_log.c \
207     ../common/__vc_pow.c \
208     ../common/__vcos.c \
209     ../common/__vcosbig.c \
210     ../common/__vcosbigf.c \
211     ../common/__vcosf.c \
212     ../common/__vexp.c \
213     ../common/__vexpf.c \
214     ../common/__vhypot.c \
215     ../common/__vhypotf.c \
216     ../common/__vlog.c \
217     ../common/__vlogf.c \
218     ../common/__vpow.c \
219     ../common/__vpowf.c \
220     ../common/__vrem_pio2m.c \
221     ../common/__vrhypot.c \
222     ../common/__vrhypotf.c \
223     ../common/__vrsqrt.c \
224     ../common/__vrsqrtf.c \
225     ../common/__vsin.c \
226     ../common/__vsinbig.c \
227     ../common/__vsinbigf.c \
228     ../common/__vsincos.c \
229     ../common/__vsincosbig.c \
230     ../common/__vsincosbigf.c \
231     ../common/__vsincosf.c \
232     ../common/__vsinf.c \
233     ../common/__vsqrt.c \
234     ../common/__vz_abs.c \
235     ../common/__vz_exp.c \
236     ../common/__vz_log.c \
237     ../common/__vz_pow.c \
238     ../common/vatan2_.c \
239     ../common/vatan2f_.c \
240     ../common/vatan_.c \
241     ../common/vatanf_.c \
242     ../common/vc_abs_.c \
243     ../common/vc_exp_.c \
244     ../common/vc_log_.c \
245     ../common/vc_pow_.c \
246     ../common/vcos_.c \
247     ../common/vcosf_.c \
248     ../common/vexp_.c \
249     ../common/vexpf_.c \
250     ../common/vhypot_.c \
251     ../common/vhypotf_.c

```

```
252     ../common/vlog_.c \
253     ../common/vlogf_.c \
254     ../common/vpow_.c \
255     ../common/vpowf_.c \
256     ../common/vrhypot_.c \
257     ../common/vrhypotf_.c \
258     ../common/vrsqrt_.c \
259     ../common/vrsqrtf_.c \
260     ../common/vsin_.c \
261     ../common/vsincos_.c \
262     ../common/vsincosf_.c \
263     ../common/vsinf_.c \
264     ../common/vsqrt_.c \
265     ../common/vsqrtf_.c \
266     ../common/vz_abs_.c \
267     ../common/vz_exp_.c \
268     ../common/vz_log_.c \
269     ../common/vz_pow_.c
```

271 .KEEP_STATE:

273 all: \$(LIBS)

275 lint: lintcheck

277 pics/%.o: ../\$(TARGET_ARCH)/src/%.S

278 \$(COMPILE.s) -o \$@ \$<

279 \$(POST_PROCESS_S_O)

277 \$(POST_PROCESS_O)

281 pics/%.o: ../common/\$(CHIP)/%.S

282 \$(COMPILE.s) -o \$@ \$<

283 \$(POST_PROCESS_S_O)

281 \$(POST_PROCESS_O)

new/usr/src/lib/libnisdb/Makefile

1

6566 Thu Apr 25 02:20:45 2019

new/usr/src/lib/libnisdb/Makefile

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 2015 Gary Mills
23 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 #
26 # Copyright (c) 2019, Joyent, Inc.
26 # Copyright (c) 2018, Joyent, Inc.
```

```
28 LIBRARY= libnisdb.a
29 VERS= .2
30 PROTOCOL_DIR= $(ROOT)/include/rpcsvc
31 YPTOL_DIR= ./yptol

33 SED= sed
```

```
35 OBJECTS = \
36 db_entry.o db_entry_c_xdr.o \
37 db_item.o db_item_c_xdr.o \
38 db_vers.o db_vers_c_xdr.o \
39 db_pickle.o \
40 db_table.o db_table_c_xdr.o \
41 db_index_entry.o db_index_entry_c.o \
42 db_index.o db_index_c_xdr.o \
43 db_scheme.o db_scheme_c_xdr.o \
44 db_query.o db_query_c_xdr.o \
45 db_mindex.o db_mindex2.o db_mindex3.o db_mindex_c_xdr.o \
46 db_log_entry.o db_log_entry_c_xdr.o \
47 db_log.o \
48 db.o \
49 xdr_nullptr.o \
50 db_dictionary.o db_dictionary_c_xdr.o \
51 db_dictxdr.o db_dictlog.o db_dictlog_c_xdr.o \
52 nis_db.o \
```

new/usr/src/lib/libnisdb/Makefile

2

```
53 nisdb_mt.o nisdb_rw.o \
54 nis_hashitem.o \
55 ldap_attr.o ldap_cto.o ldap_glob.o ldap_ldap.o \
56 ldap_map.o ldap_nisdbquery.o \
57 ldap_op.o ldap_parse.o ldap_print.o ldap_ruleval.o \
58 ldap_util.o ldap_val.o ldap_xdr.o ldap_scheme.o \
59 nis_ldap.o \
60 nis_parse_ldap_attr.o nis_parse_ldap_conf.o \
61 nis_parse_ldap_map.o nis_parse_ldap_util.o \
62 nis_parse_ldap_yp_util.o \
63 dit_access.o \
64 dit_access_utils.o \
65 lock_update.o \
66 map_conv.o \
67 map_ctrl.o \
68 yptol_utils.o \
69 shim.o \
70 shim_ancil.o \
71 shim_lockmap.o \
72 ttl_utils.o \
73 update.o \
74 shim_changepasswd.o \
75 stubs.o

77 X_FILES= \
78 db_c.x db_dictionary_c.x db_entry_c.x db_index_c.x \
79 db_index_entry_c.x db_item_c.x db_log_c.x db_log_entry_c.x \
80 db_mindex_c.x db_query_c.x db_scheme_c.x db_table_c.x \
81 db_vers_c.x db_dictlog_c.x

83 DERIVED_HEADERS= $(X_FILES:%_c.x=%_c.h) $(X_FILES:%_c.x=%_h)
84 DERIVED_SOURCES= $(X_FILES:%_c.x=%_c_xdr.c)
85 DERIVED_FILES= $(DERIVED_HEADERS) $(DERIVED_SOURCES)

87 # delete the derived files when cleaning up
88 CLEANFILES += $(DERIVED_FILES)

90 # include library definitions
91 include ../Makefile.lib

93 MAPFILES = mapfile-vers

95 SRCS= db_dictionary_c_xdr.c db_dictlog_c_xdr.c db_dictxdr.c \
96 db_entry_c_xdr.c db_index_c_xdr.c db_index_entry_c_xdr.c db_item_c_xdr.c \
97 db_log_c_xdr.c db_log_entry_c_xdr.c db_mindex_c_xdr.c db_query_c_xdr.c \
98 db_scheme_c_xdr.c db_table_c_xdr.c db_vers_c_xdr.c \
99 $(YPTOL_DIR)/dit_access.c \
100 $(YPTOL_DIR)/dit_access_utils.c \
101 $(YPTOL_DIR)/lock_update.c \
102 $(YPTOL_DIR)/map_conv.c \
103 $(YPTOL_DIR)/map_ctrl.c \
104 $(YPTOL_DIR)/yptol_utils.c \
105 $(YPTOL_DIR)/shim.c \
106 $(YPTOL_DIR)/shim_ancil.c \
107 $(YPTOL_DIR)/shim_lockmap.c \
108 $(YPTOL_DIR)/ttl_utils.c \
109 $(YPTOL_DIR)/update.c \
110 $(YPTOL_DIR)/shim_changepasswd.c \
111 $(YPTOL_DIR)/stubs.c \
112 #db_c_xdr.c xdr_nullptr.c

114 # Libnisdb has grown large enough that there may be too many symbols
115 # wanting 'small' PIC references. Hence, compile some files with
116 # 'large' PIC references; the parser files are good candidates, since
117 # they're only used at startup, and performance isn't too important.
118 LARGESYMS= nis_parse_ldap_conf.o \
```

new/usr/src/lib/libnisdb/Makefile

3

```

119         nis_parse_ldap_attr.o \
120         nis_parse_ldap_yp_util.o \
121         nis_parse_ldap_map.o \
122         ldap_parse.o \
123         dit_access.o \
124         dit_access_utilis.o \
125         map_conv.o \
126         map_ctrl.o \
127         shim.o \
128         shim_ancil.o \
129         shim_changepasswd.o \
130         update.o \
131         stubs.o \
132         yptol_utils.o \
133         nis_parse_ldap_util.o
134 LARGEPICS= $(LARGESYMS:%=pics/)
135 $(LARGEPICS) := sparc_C_PICFLAGS = $(C_BIGPICFLAGS)
136 $(LARGEPICS) := i386_C_PICFLAGS = $(C_BIGPICFLAGS)

138 # More and stricter semantic checks and lint-like checks
139 CFLAGS += $(CCVERBOSE)

141 CPPFLAGS += -D_REENTRANT

143 # Have an unused variable that can't be removed
144 UVAR_PICS = \
145 pics/db_dictionary_c_xdr.o \
146 pics/db_dictlog_c_xdr.o \
147 pics/db_entry_c_xdr.o \
148 pics/db_index_c_xdr.o \
149 pics/db_item_c_xdr.o \
150 pics/db_log_entry_c_xdr.o \
151 pics/db_mindex_c_xdr.o \
152 pics/db_query_c_xdr.o \
153 pics/db_scheme_c_xdr.o \
154 pics/db_table_c_xdr.o \
155 pics/db_vers_c_xdr.o \
156 pics/ldap_xdr.o \
157 pics/dit_access.o \
158 pics/update.o \
159 pics/yptol_utils.o \
160 pics/map_ctrl.o

162 $(UVAR_PICS) := CERRWARN += -_gcc=-Wno-unused-variable

164 # Correcting these might alter logic
165 CERRWARN += -_gcc=-Wno-uninitialized
166 CERRWARN += -_gcc=-Wno-switch

168 # not linted
169 SMATCH=off

171 # Extra includes, from yp, for yptol files.
172 CPPFLAGS += -I$(SRC)/cmd/ypcmd

174 # Need the path to nis_clnt.h
175 CLNT_PICS = pics/db_entry_c_xdr.o pics/ldap_xdr.o pics/db_mindex_c_xdr.o

177 $(CLNT_PICS) := CPPFLAGS += -I../libnsl/nis/gen

179 LIBS = $(DYNLIB)
180 ZDEFS=
181 LDLIBS += -lnsl -lldap -lc

183 # A number of interfaces are interposed by numerous applications, therefore
184 # prevent direct binding to anything in libnisdb. Disable libnisdb from

```

new/usr/src/lib/libnisdb/Makefile

4

```

185 # directly binding to itself, but allow libnisdb to directly bind to its
186 # dependencies (ie. map -Bdirect -> -zdirect). Ensure lazy loading is
187 # established (which is enabled automatically with -Bdirect). In addition,
188 # libnisdb interposes on socket(), so tag this library as an interposer.
189 # dependencies (lazyload).
190 BDIRECT =
191 DYNFLAGS += $(BNODIRECT) $(ZINTERPOSE) $(ZDIRECT) $(ZLAZYLOAD)

193 all := TARGET= all
194 clean := TARGET= clean
195 clobber := TARGET= clobber
196 install := TARGET= install
197 lint := TARGET= lint

199 .KEEP_STATE:

201 all: $(DERIVED_FILES) $(LIBS)

203 install: all $(ROOTLIBS) $(ROOTLINKS)

205 lint: $(DERIVED_FILES) .WAIT lintcheck

207 pics/%.o: %.c
208     $(COMPILE.c) -o $@ $<
209     $(POST_PROCESS_O)

211 pics/%.o: %.cc
212     $(COMPILE.cc) -o $@ $<
213     $(POST_PROCESS_O)

207 objs/%.o pics/%.o profs/%.o: $(YPTOL_DIR)/%.c
208     $(COMPILE.c) -o $@ $<
209     $(POST_PROCESS_O)

211 #
212 # Pattern matching rules that define how to build the derived files.
213 #
214 %_c.h: %_c.x
215     $(RM) $@
216     $(RPCGEN) -DUSINGC -h -o $@ $<

218 %_h: %_c.x
219     $(RM) $@
220     $(RPCGEN) -h -o $@ $<

222 %_c_xdr.c: %_c.x
223     $(RM) $@
224     $(RPCGEN) -DUSINGC -c -o $@ $<

226 # rename the xdr_db_free_entry() in the db_table_c_xdr.c. So the one
227 # in db_index_entry_c is used instead.
228 DB_TABLE=db_table_c

230 $(DB_TABLE)_xdr.c: $(DB_TABLE).x
231     $(RM) $@
232     $(RPCGEN) -DUSINGC -c $(DB_TABLE).x | \
233     $(SED) -e 's/^xdr_db_free_entry(/__OBSOLETEDEXdr_db_free_entry(/' > $@

235 # include library targets
236 include ../Makefile.targ

```

new/usr/src/lib/libsqlite/Makefile.com

1

6690 Thu Apr 25 02:20:46 2019

new/usr/src/lib/libsqlite/Makefile.com

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
1 #
2 # Copyright 2008 Sun Microsystems, Inc. All rights reserved.
3 # Use is subject to license terms.
4 # Copyright 2015 Igor Kozhukhov <ikozhukhov@gmail.com>
5 # Copyright 2016 Nexenta Systems, Inc. All rights reserved.
6 # Copyright (c) 2019, Joyent, Inc.
6 # Copyright (c) 2018, Joyent, Inc.
7 # Copyright 2019 OmniOS Community Edition (OmniOSce) Association.
8 #
```

```
10 # Make the SO name unlikely to conflict with any other
11 # libsqlite that might also be found on the system.
12 LIBRARY = libsqlite-sys.a
```

```
14 VERS = .2.8.15
15 OBJECTS = \
16     attach.o \
17     auth.o \
18     btree.o \
19     btree_rb.o \
20     build.o \
21     copy.o \
22     date.o \
23     delete.o \
24     encode.o \
25     expr.o \
26     func.o \
27     hash.o \
28     insert.o \
29     main.o \
30     opcodes.o \
31     os.o \
32     pager.o \
33     parse.o \
34     pragma.o \
35     printf.o \
36     random.o \
37     select.o \
38     table.o \
39     tokenize.o \
40     trigger.o \
41     update.o \
42     util.o \
43     vacuum.o \
44     vdbe.o \
45     vdbeaux.o \
46     where.o
```

```
48 include $(SRC)/lib/Makefile.lib
```

```
50 # install this library in the root filesystem
51 include $(SRC)/lib/Makefile.rootfs
```

new/usr/src/lib/libsqlite/Makefile.com

2

```
53 SRCDIR = ../src
54 TOOLDIR = ../tool
55 $(DYNLIB) := LDLIBS += -lc
56 LIBS = $(DYNLIB) $(LINTLIB) $(NATIVERELOC)
```

```
58 $(LINTLIB) := SRCS = ../$(LINTSRC)
```

```
60 # generated sources
61 GENSRCS = opcodes.c parse.c
```

```
63 # all sources
```

```
64 SRCS = \
65     $(GENSRC) \
66     $(SRCDIR)/attach.c \
67     $(SRCDIR)/auth.c \
68     $(SRCDIR)/btree.c \
69     $(SRCDIR)/btree_rb.c \
70     $(SRCDIR)/build.c \
71     $(SRCDIR)/copy.c \
72     $(SRCDIR)/date.c \
73     $(SRCDIR)/delete.c \
74     $(SRCDIR)/encode.c \
75     $(SRCDIR)/expr.c \
76     $(SRCDIR)/func.c \
77     $(SRCDIR)/hash.c \
78     $(SRCDIR)/insert.c \
79     $(SRCDIR)/main.c \
80     $(SRCDIR)/os.c \
81     $(SRCDIR)/pager.c \
82     $(SRCDIR)/pragma.c \
83     $(SRCDIR)/printf.c \
84     $(SRCDIR)/random.c \
85     $(SRCDIR)/select.c \
86     $(SRCDIR)/table.c \
87     $(SRCDIR)/tokenize.c \
88     $(SRCDIR)/update.c \
89     $(SRCDIR)/util.c \
90     $(SRCDIR)/vacuum.c \
91     $(SRCDIR)/vdbe.c \
92     $(SRCDIR)/vdbeaux.c \
93     $(SRCDIR)/where.c \
94     $(SRCDIR)/trigger.c
```

```
96 MYCPPFLAGS = -D_REENTRANT -DTHREADSAFE=1 -DHAVE_USLEEP=1 -I. -I.. -I$(SRCDIR)
97 CPPFLAGS += $(MYCPPFLAGS)
```

```
99 CERRWARN += -_gcc=-Wno-implicit-function-declaration
100 CERRWARN += -_gcc=-Wno-uninitialized
101 CERRWARN += -_gcc=-Wno-unused-function
102 CERRWARN += -_gcc=-Wno-unused-label
```

```
104 # not linted
105 SMATCH=off
```

```
107 MAPFILES = ../mapfile-sqlite
```

```
109 # headers generated here
110 GENHDR = opcodes.h parse.h
```

```
112 # Header files used by all library source files.
```

```
113 #
114 HDR = \
115     $(GENHDR) \
116     $(SRCDIR)/btree.h \
117     $(SRCDIR)/config.h \
118     $(SRCDIR)/hash.h \
```


new/usr/src/lib/libsqlite/Makefile.com

3

```

119 $(SRCDIR)/os.h \
120 ../sqlite.h \
121 $(SRCDIR)/sqliteInt.h \
122 $(SRCDIR)/vdbe.h \
123 $(SRCDIR)/vdbeInt.h

125 #
126 # Sources used for test harness
127 #
128 TESTSRC = \
129 $(SRCDIR)/tclsqlite.c \
130 $(SRCDIR)/btree.c \
131 $(SRCDIR)/func.c \
132 $(SRCDIR)/os.c \
133 $(SRCDIR)/pager.c \
134 $(SRCDIR)/test1.c \
135 $(SRCDIR)/test2.c \
136 $(SRCDIR)/test3.c \
137 $(SRCDIR)/md5.c

139 TESTOBJS = $(TESTSRC:$(SRCDIR)/%.c=%o)

141 TESTCLEAN = $(TESTOBJS) test.db test.tcl test1.bt test2.db testdb

143 #
144 # Native variant (needed by cmd/configd)
145 #
146 NATIVERELOC = libsqlite-native.o
147 NATIVEPROGS = testfixture
148 NATIVEOBJS = $(OBJS:%.o=%-native.o)

150 NATIVETARGETS = $(NATIVEPROGS) $(NATIVEOBJS) $(NATIVERELOC)

152 $(NATIVETARGETS) := CC = $(NATIVECC)
153 $(NATIVETARGETS) := LD = $(NATIVELD)
154 $(NATIVETARGETS) := CFLAGS = $(NATIVE_CFLAGS)
155 $(NATIVETARGETS) := CPPFLAGS = $(MYCPPFLAGS)
156 $(NATIVETARGETS) := LDFLAGS =
157 $(NATIVETARGETS) := LDLIBS = -lc

159 $(OBJS) := CFLAGS += $(CTF_FLAGS)
160 $(OBJS) := CTFCONVERT_POST = $(CTFCONVERT_O)
161 $(NATIVEOBJS) := CFLAGS += $(CTF_FLAGS)
162 $(NATIVEOBJS) := CTFCONVERT_POST = $(CTFCONVERT_O)

164 TCLBASE = /usr/sfw
165 TCLVERS = tcl8.3

167 testfixture := MYCPPFLAGS += -I$(TCLBASE)/include -DTCLSH -DSQLITE_TEST=1
168 #
169 # work around compiler issues
170 #
171 testfixture := CFLAGS += \
172 -erroff=E_ARRAY_OF_INCOMPLETE \
173 -erroff=E_ARG_INCOMPATIBLE_WITH_ARG

175 testfixture := LDLIBS += -R$(TCLBASE)/lib -L$(TCLBASE)/lib -l$(TCLVERS) -lm -ldl

177 CLEANFILES += \
178 $(NATIVETARGETS) \
179 $(TESTCLEAN) \
180 lemon \
181 lemon.o \
182 lempar.c \
183 opcodes.c \
184 opcodes.h

```

new/usr/src/lib/libsqlite/Makefile.com

4

```

185 parse_tmp.c \
186 parse_tmp.h \
187 parse_tmp.out \
188 parse_tmp.y \
189 parse.c \
190 parse.h

192 ENCODING = ISO8859

195 .PARALLEL: $(OBJS) $(OBJS:%.o=%-native.o)
196 .KEEP_STATE:

198 # This is the default Makefile target. The objects listed here
199 # are what get build when you type just "make" with no arguments.
200 #
201 all: $(LIBS)
202 install: all \
203 $(ROOTLIBDIR)/$(DYNLIB) \
204 $(ROOTLIBDIR)/$(NATIVERELOC)

206 lint:

208 all_h: $(GENHDR)

210 $(ROOTLIBDIR)/$(NATIVERELOC) := FILEMODE= 644
211 $(ROOTLINTDIR)/$(LINTLIB) := FILEMODE= 644

213 $(ROOTLINK): $(ROOTLIBDIR) $(ROOTLIBDIR)/$(DYNLIB)
214 $(INS.liblink)

216 $(ROOTLINTDIR)/%: ../%
217 $(INS.file)

219 native: $(NATIVERELOC)

221 $(NATIVERELOC): objs .WAIT $(OBJS:%.o=%-native.o)
222 $(LD) -r -o $(NATIVERELOC) $(OBJS:%.o=%-native.o)

224 opcodes.h: $(SRCDIR)/vdbe.c
225 @echo "Generating $@"; \
226 $(RM) -f $@; \
227 echo '/* Automatically generated file. Do not edit */' > $@; \
228 grep '^case OP_' $(SRCDIR)/vdbe.c | \
229 sed -e 's/:/ /' | \
230 $(AWK) '{printf "#define %-30s %3d\n", $$2, ++cnt}' >> $@

232 opcodes.c: $(SRCDIR)/vdbe.c
233 @echo "Generating $@"; \
234 $(RM) -f $@; \
235 echo '/* Automatically generated file. Do not edit */' > $@; \
236 echo 'char *sqliteOpcodeNames[] = { "???", ' >> $@; \
237 grep '^case OP_' $(SRCDIR)/vdbe.c | \
238 sed -e 's/^. *OP_/' -e 's/:.*$$/' -e 's/ /' >> $@; \
239 echo '};' >> $@

241 testfixture: FRC
242 @if [ -f $(TCLBASE)/include/tcl.h ]; then \
243 unset SUNPRO_DEPENDENCIES; \
244 echo $(LINK.c) -o testfixture $(TESTSRC) $(LIBRARY) $(LDLIBS) ;\
245 exec $(LINK.c) -o testfixture $(TESTSRC) $(LIBRARY) $(LDLIBS) ;\
246 else \
247 echo "$(TCLBASE)/include/tcl.h: not found."; \
248 exit 1; \
249 fi

```

```
251 # Prevent Makefile.lib $(PICS) := from adding PICFLAGS
252 # by building lemon in a recursive make invocation.
253 # Otherwise, this target causes a rebuild every time after
254 # the PICS target builds this one way, then lint the other.
255 parse.h parse.c : $(SRCDIR)/parse.y $(TOOLDIR)/lemon.c $(TOOLDIR)/lempar.c
256     -$(RM) parse_tmp.y lempar.c
257     $(CP) $(SRCDIR)/parse.y parse_tmp.y
258     $(CP) $(TOOLDIR)/lempar.c lempar.c
259     $(MAKE) lemon
260     ./lemon parse_tmp.y
261     -$(RM) parse.c parse.h
262     $(CP) parse_tmp.h parse.h
263     $(CP) parse_tmp.c parse.c

265 lemon: $(TOOLDIR)/lemon.c
266     $(NATIVECC) $(NATIVE_CFLAGS) -o $@ $(TOOLDIR)/lemon.c

268 objs/%-native.o: $(SRCDIR)/%.c $(GENHDR)
269     $(COMPILE.c) -o $@ $<
270     $(POST_PROCESS_O)

272 objs/%-native.o: %.c $(GENHDR)
273     $(COMPILE.c) -o $@ $<
274     $(POST_PROCESS_O)

276 objs/parse-native.o: parse.c $(GENHDR)
277     $(COMPILE.c) -o $@ parse.c
278     $(POST_PROCESS_O)

280 objs/%.o pics/%.o: $(SRCDIR)/%.c $(GENHDR)
281     $(COMPILE.c) -o $@ $<
282     $(POST_PROCESS_O)

284 objs/%.o pics/%.o: %.c $(GENHDR)
285     $(COMPILE.c) -o $@ $<
286     $(POST_PROCESS_O)

288 # need direct rules for generated files
289 objs/opcodes.o pics/opcodes.o: opcodes.c $(GENHDR)
290     $(COMPILE.c) -o $@ opcodes.c
291     $(POST_PROCESS_O)

293 objs/parse.o pics/parse.o: parse.c $(GENHDR)
294     $(COMPILE.c) -o $@ parse.c
295     $(POST_PROCESS_O)

297 include $(SRC)/lib/Makefile.targ

299 FRC:
```

```

*****
2431 Thu Apr 25 02:20:46 2019
new/usr/src/lib/libtfnfctl/Makefile.com
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 2006 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
25 # Copyright (c) 2019, Joyent, Inc.
25 # Copyright (c) 2018, Joyent, Inc.

27 LIBRARY=      libtfnfctl.a
28 VERS=         .1
29 OBJECTS.c=    open.o      \
30               prb_child.o  \
31               prb_shmem.o  \
32               prb_proc.o   \
33               prb_lmap.o   \
34               prb_rtl.d.o  \
35               prb_findexec.o \
36               prb_status.o \
37               util.o       \
38               traverse.o   \
39               sym.o        \
40               elf.o        \
41               continue.o  \
42               checklib.o  \
43               probes.o    \
44               probes_ext.o \
45               close.o     \
46               targmem.o  \
47               comb.o      \
48               kernel_int.o \
49               kernel.o    \
50               internal.o  \
51               status.o

```

```

54 OBJECTS.s=    $(MACH)_asm.o
56 OBJECTS=      $(OBJECTS.c) $(OBJECTS.s)

58 include ../..//Makefile.lib

60 # We omit $(OBJECTS.s:%.o=%.s) in the next line, because lint no like
61 SRCS= $(OBJECTS.c:%.o=../%.c)

63 LIBS=         $(DYNLIB)

65 HDRS=        tnfctl.h
66 Roothdrdir=  $(ROOT)/usr/include/tnf
67 Roothdrs=    $(HDRS:%=$(Roothdrdir)/%)
68 CHECKHDRS=   $(HDRS:%.h=%.check)
69 $(Roothdrs) := FILEMODE = 0644
70 CHECKHDRS =  $(HDRS:%.h=%.check)

72 LDLIBS +=    -lc -lelf

74 CPPFLAGS +=  -I$(SRC)/lib/libtnfprobe -D_REENTRANT -I$(SRC)/cmd/sgs/include

76 LINTFLAGS += -y

78 CERRWARN +=  -_gcc=-Wno-uninitialized
79 CERRWARN +=  -_gcc=-Wno-empty-body
80 CERRWARN +=  -_gcc=-Wno-parentheses

82 # not linted
83 SMATCH=off

85 ASFLAGS +=   -P

87 $(Roothdrs) := FILEMODE = 644

89 .KEEP_STATE:

91 all: $(LIBS)

93 install_h: $(Roothdrdir) $(Roothdrs)

95 lint:
96     $(LINT.c) $(SRCS)

98 check: $(CHECKHDRS)

100 $(ROOTLIBDIR) $(Roothdrdir):
101     $(INS.dir)

103 $(Roothdrdir)/% : %
104     $(INS.file)

106 BUILD.s=    $(AS) $< -o $@

108 objs/%.o pics/%.o: ../%.s
109     $(COMPILE.s) -o $@ $<
110     $(POST_PROCESS_S_O)
110     $(POST_PROCESS_O)

112 objs/%.o pics/%.o: ../%.c
113     $(COMPILE.c) -o $@ $<
114     $(POST_PROCESS_O)

117 %.o:        ../%.c

```

new/usr/src/lib/libtnftl/Makefile.com

3

```
118      $(COMPILE.c) -o $@ $<  
119      $(POST_PROCESS_O)  
  
121 include ../../Makefile.targ
```

new/usr/src/lib/libtnfprobe/Makefile.com

1

```
*****
2616 Thu Apr 25 02:20:46 2019
new/usr/src/lib/libtnfprobe/Makefile.com
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 (c) 1997, 2010, Oracle and/or its affiliates. All rights reserved.
23 #
24 # Copyright (c) 2019, Joyent, Inc.
24 # Copyright (c) 2018, Joyent, Inc.

26 LIBRARY=      libtnfprobe.a
27 VERS=         .1
28 OBJECTS.c=    tnf_buf.o \
29               trace_init.o \
30               trace_funcs.o \
31               debug_funcs.o \
32               probe_mem.o \
33               tnf_args.o \
34               tnf_trace.o \
35               probe_cntl.o

37 UFSODIR=      $(SRC)/uts/common/tnf
38 UFSOBSJS=     tnf_writer.o tnf_probe.o

40 OBJECTS.s=    $(MACH)_locks.o

42 OBJECTS=      $(OBJECTS.c) $(UFSOBSJS) $(OBJECTS.s)

44 include ../../Makefile.lib

46 # We omit $(OBJECTS.s:%.o=%.s) in the next line, because lint no like
47 SRCS= $(OBJECTS.c:%.o=../%.c) $(UFSOBSJS:%.o=$(UFSODIR)/%.c)

49 LIBS=        $(DYNLIB)

51 DYNFLGAS +=   $(ZINTERPOSE)
```

new/usr/src/lib/libtnfprobe/Makefile.com

2

```
53 HDRS=        com.h writer.h probe.h
54 ROOTHDRDIR=  $(ROOT)/usr/include/tnf
55 ROOTHDRS=    $(HDRS:%=$(ROOTHDRDIR)/%)
56 CHECKHDRS=   $(HDRS:%.h=%.check)
57 $(ROOTHDRS) := FILEMODE = 0644
58 CHECKHDRS =  $(HDRS:%.h=%.check)

60 # Include .. first to pick up tnf_trace.h in current dir, Include UFSODIR to
61 #           pick up tnf_types.h
62 CPPFLGAS +=  -I.. -I$(UFSODIR) -D_REENTRANT -D_TNF_LIBRARY

64 LINTFLGAS += -y

66 CERRWARN +=  -_gcc=-Wno-unused-variable
67 CERRWARN +=  -_gcc=-Wno-parentheses
68 CERRWARN +=  -_gcc=-Wno-uninitialized

70 # not linted
71 SMATCH=off

73 $(ROOTHDRS) := FILEMODE = 644

75 LDLIBS +=   -lc

77 .KEEP_STATE:

79 all: $(LIBS)

81 install_h: $(ROOTHDRDIR) $(ROOTHDRS)

83 lint:
84         $(LINT.c) $(SRCS)

86 check: $(CHECKHDRS)

88 $(ROOTLIBDIR) $(ROOTHDRDIR):
89         $(INS.dir)

91 $(ROOTHDRDIR)/% : %
92         $(INS.file)

94 #ASFLGAS=     $(AS_PICFLGAS) -P -D_SYS_SYS_S -D_LOCORE -D_ASM -DPIC -DLOCORE $
95 ASFLGAS=      -P -D_SYS_SYS_S -D_LOCORE -D_ASM -DPIC -DLOCORE $(CPPFLGAS)
96 BUILD.s=     $(AS) $(ASFLGAS) $< -o $@

98 objs/%.o pics/%.o: ../%.s
99             $(COMPILE.s) -o $@ $<
100             $(POST_PROCESS_S_O)
100             $(POST_PROCESS_O)

102 pics/%.o objs/%.o: ../%.c
103             $(COMPILE.c) -o $@ $<
104             $(POST_PROCESS_O)

106 pics/%.o objs/%.o: $(UFSODIR)/%.c
107             $(COMPILE.c) -o $@ $<
108             $(POST_PROCESS_O)

110 include ../../Makefile.targ
```

new/usr/src/lib/libumem/Makefile.targ

1

2375 Thu Apr 25 02:20:46 2019

new/usr/src/lib/libumem/Makefile.targ

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

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 2007 Sun Microsystems, Inc. All rights reserved.

23 # Use is subject to license terms.

24 # Copyright (c) 2016, Chris Fraire <cfraire@me.com>.

25 # Copyright (c) 2019, Joyent, Inc.

26 #

28 #

29 # We build each flavor in a separate make invocation to improve clarity(!) in

30 # Makefile.com. The subordinate makes have \$(CURTYPE) set to indicate the

31 # flavor they're supposed to build. This causes the correct set of source

32 # files and compiler and linker flags to be selected.

33 #

35 install: \$(TYPES:%=install.%)

37 install.library: all.library \$(INSTALL_DEPS_library)

38 install.standalone: all.standalone \$(INSTALL_DEPS_standalone)

40 all: \$(TYPES:%=all.%)

42 \$(TYPES:%=all.%):

43 @\$(MAKE) \$@.targ CURTYPE=\$(@:all.%=%)

45 all.library.targ: \$(LIBS)

46 all.standalone.targ: \$(STANDLIBRARY)

48 lint: \$(TYPES:%=lint.%)

50 \$(TYPES:%=lint.%):

51 @\$(MAKE) lintcheck CURTYPE=\$(@:lint.%=%)

53 \$(STANDLIBRARY): \$(OBJS) \$(LINKTEST_OBJ)

new/usr/src/lib/libumem/Makefile.targ

2

54 \$(LD) \$(BREDUCE) \$(ZDEFS) \$(LDLFLAGS) -o \$@.linktest \$(OBJS) \$(LINKTEST_

55 rm \$@.linktest

56 \$(LD) \$(LDLFLAGS) -o \$@ \$(OBJS)

58 clobber: \$(TYPES:%=clobber.%)

60 \$(TYPES:%=clobber.%):

61 @\$ (MAKE) clobber.targ CURTYPE=\$(@:clobber.%=%)

63 clobber.targ: clean

64 -\$(RM) \$(CLOBBERTARGETFILES)

66 # include library targets

67 include ../../Makefile.targ

69 \$(PICS): pics

70 \$(OBJS): objs

72 objs/%.o pics/%.o: \$(ISASRCDIR)/%.c

73 \$(COMPILE.c) -o \$@ \$<

74 \$(POST_PROCESS_O)

76 objs/%.o pics/%.o: \$(ISASRCDIR)/%.s

77 \$(COMPILE.s) -o \$@ \$<

78 \$(POST_PROCESS_S_O)

77 \$(POST_PROCESS_O)

80 objs/%.o pics/%.o: ../common/%.c

81 \$(COMPILE.c) -o \$@ \$<

82 \$(POST_PROCESS_O)

84 # install rule for lint library target

85 \$(ROOTLINTDIR)/%: ../common/%

86 \$(INS.file)

new/usr/src/lib/pkcs11/libsoftcrypto/amd64/Makefile

1

3038 Thu Apr 25 02:20:47 2019

new/usr/src/lib/pkcs11/libsoftcrypto/amd64/Makefile

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 #
23 # Copyright (c) 2008, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright 2017 Jason King
25 # Copyright (c) 2019, Joyent, Inc.
26 # Copyright (c) 2018, Joyent, Inc.
27 # Copyright (c) 2018, Joyent, Inc.
28 LIBRARY = libsoftcrypto.a
29 VERS = .1
30 #
31 include ../Makefile.com
32 #
33 AES_PSM_OBJS = aes_amd64.o aes_intel.o aeskey.o
34 ARCFOUR_PSM_OBJS = arcfour-x86_64.o
35 BIGNUM_PSM_OBJS = bignum_amd64.o bignum_amd64_asm.o
36 MODES_PSM_OBJS = gcm_intel.o
37 #
38 include $(SRC)/lib/Makefile.lib
39 include $(SRC)/lib/Makefile.lib.64
40 #
41 CERRWARN += -_gcc=-Wno-type-limits
42 CERRWARN += -_gcc=-Wno-parentheses
43 CERRWARN += -_gcc=-Wno-uninitialized
44 #
45 # not linted
46 SMATCH=off
47 #
48 AES_PSM_SRC = $(AES_DIR)/$(MACH64)/aes_amd64.s \
49 $(AES_DIR)/$(MACH64)/aes_intel.s \
50 $(AES_DIR)/$(MACH64)/aeskey.c
51 ARCFOUR_PSM_SRC = arcfour-x86_64.s
```

new/usr/src/lib/pkcs11/libsoftcrypto/amd64/Makefile

2

```
52 BIGNUM_PSM_SRC = $(BIGNUM_DIR)/$(MACH64)/bignum_amd64.c \
53 $(BIGNUM_DIR)/$(MACH64)/bignum_amd64_asm.s
54 #
55 MODES_PSM_SRC = $(MODES_DIR)/$(MACH64)/gcm_intel.s
56 #
57 # Sources need to be redefined after Makefile.lib inclusion.
58 SRCS = $(AES_SRC) $(ARCFOUR_SRC) $(BIGNUM_SRC) $(BLOWFISH_SRC) \
59 $(DES_SRC) $(MODES_SRC) $(DH_SRC) $(DSA_SRC) $(RSA_SRC) \
60 $(PAD_SRC)
61 #
62 SRCDIR = $(SRC)/lib/pkcs11/libsoftcrypto/common
63 #
64 LIBS = $(DYNLIB) $(LINTLIB)
65 MAPFILEDIR = ../common
66 #
67 CFLAGS += -xO4 -xcrossfile
68 CPPFLAGS += -I$(CRYPTODIR) -I$(UTSDIR) -D_POSIX_PTHREAD_SEMANTICS
69 ASFLAGS += $(AS_PICFLAGS) -P -D__STDC__ -D_ASM
70 BIGNUM_FLAGS += -DPSR_MUL
71 CLEANFILES += arcfour-x86_64.s
72 LDLIBS += -lcryptoutil -lc
73 LINTFLAGS64 += $(EXTRA_LINT_FLAGS)
74 #
75 .KEEP_STATE:
76 #
77 all: $(LIBS)
78 #
79 lint: $(SRCS)
80 #
81 install: all $(ROOTLIBS64) $(ROOTLINKS64) $(ROOTLINT64)
82 #
83 pics/%.o: $(AES_DIR)/$(MACH64)/%.c
84 $(COMPILE.c) $(AES_FLAGS) -o $@ $<
85 $(POST_PROCESS_O)
86 #
87 pics/%.o: $(AES_DIR)/$(MACH64)/%.s
88 $(COMPILE.s) $(AES_FLAGS) -o $@ $<
89 $(POST_PROCESS_S_O)
90 $(POST_PROCESS_O)
91 #
92 pics/%.o: $(BIGNUM_DIR)/$(MACH64)/%.c
93 $(COMPILE.c) $(BIGNUM_FLAGS) -o $@ $<
94 $(POST_PROCESS_O)
95 #
96 pics/%.o: $(BIGNUM_DIR)/$(MACH64)/%.s
97 $(COMPILE64.s) $(BIGNUM_FLAGS) -o $@ $<
98 $(POST_PROCESS_S_O)
99 $(POST_PROCESS_O)
100 #
101 pics/%.o: $(MODES_DIR)/$(MACH64)/%.s
102 $(COMPILE64.s) $(MODES_FLAGS) -o $@ $<
103 $(POST_PROCESS_S_O)
104 $(POST_PROCESS_O)
105 #
106 include ../Makefile.targ
107 #
108 arcfour-x86_64.s: $(ARCFOUR_DIR)/amd64/arcfour-x86_64.pl
109 $(PERL) $? $@
110 #
111 pics/%.o: arcfour-x86_64.s
112 $(COMPILE64.s) $(ARCFOUR_FLAGS) -o $@ $<
113 $(POST_PROCESS_S_O)
114 $(POST_PROCESS_O)
```

new/usr/src/lib/pkcs11/libsoftcrypto/capabilities/intel/i386/Makefile 1

```
*****
1772 Thu Apr 25 02:20:47 2019
new/usr/src/lib/pkcs11/libsoftcrypto/capabilities/intel/i386/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 #
23 # Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
24 #
25 # Copyright (c) 2019, Joyent, Inc.
26 #

28 BIGNUM_PSM_OBJS =      bignum_i386.o bignum_i386_asm.o

30 include      ../../Makefile.com

32 # Redefine the objects required for this capabilities group.
33 OBJECTS =    $(BIGNUM_OBJS)

35 include      $(SRC)/lib/Makefile.lib

37 AS_CPPFLAGS +=  -D__STDC__ -D_ASM -DPIC -D_REENTRANT -D$(MACH)
38 ASFLAGS +=     $(AS_PICFLAGS) -P
39 CFLAGS +=      $(CVERBOSE)
40 CPPFLAGS +=    -I$(CRYPTODIR) -I$(UTSDIR) -D_POSIX_PTHREAD_SEMANTICS
41 BIGNUM_FLAGS += -DMMX_MANAGE -DHWCAP -DPSR_MUL -DNO_BIG_ONE -DNO_BIG_TWO

43 CERRWARN +=    -_gcc=-Wno-unused-variable

45 .KEEP_STATE:

47 all install:  $(SYMCAP)

49 include      ../../Makefile.targ

51 pics/bignum_i386.o: $(BIGNUM_DIR)/i386/bignum_i386.c
52                   $(COMPILE.c) $(BIGNUM_FLAGS) -o $@ \
53                   $(BIGNUM_DIR)/i386/bignum_i386.c
```

new/usr/src/lib/pkcs11/libsoftcrypto/capabilities/intel/i386/Makefile 2

```
54                   $(POST_PROCESS_O)

56 pics/bignum_i386_asm.o: $(BIGNUM_DIR)/i386/bignum_i386_asm.s
57                   $(COMPILE.s) -o $@ $(BIGNUM_DIR)/i386/bignum_i386_asm.s
58                   $(POST_PROCESS_S_O)
59                   $(POST_PROCESS_O)
```


new/usr/src/test/crypto-tests/tests/common/Makefile

1

926 Thu Apr 25 02:20:47 2019

new/usr/src/test/crypto-tests/tests/common/Makefile

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 at
9 # http://www.illumos.org/license/CDDL.
10 #
```

```
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2015 Nexenta Systems, Inc. All rights reserved.
15 # Copyright (c) 2019, Joyent, Inc.
16 #
```

```
18 include $(SRC)/cmd/Makefile.cmd
19 include $(SRC)/cmd/Makefile.ctf
20 include $(SRC)/test/Makefile.com
```

```
22 OBJS = testfuncs.o cryptotest_pkcs.o cryptotest_kcf.o
23 SRCS = $(OBJS:%.o=%.c)
```

```
25 CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64 -I$(SRC)/common/crypto/
```

```
27 all install: $(OBJS)
```

```
29 %.o: %.c
30     $(COMPILE.c) -o $@ $<
31     $(POST_PROCESS_O)
```

```
33 lint:
```

```
35 clobber clean:
36     -$(RM) $(OBJS)
```

new/usr/src/test/util-tests/runfiles/default.run

1

1445 Thu Apr 25 02:20:47 2019

new/usr/src/test/util-tests/runfiles/default.run

10816 ctf_dwarf_convert_type() relies on un-initialized id

10817 ctfconvert -i option is mis-handled

10818 Improve ctfconvert error messages

10819 ctfconvert should handle empty dies

10820 ctfconvert -i never converts

10821 bad free in ctf_dwarf_init_die

10815 shouldn't build gcore.c as part of kmdb

Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>

```
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 at
9 # http://www.illumos.org/license/CDDL.
10 #
```

```
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 # Copyright 2014 Nexenta Systems, Inc. All rights reserved.
16 # Copyright 2019, Joyent, Inc.
16 # Copyright 2017 Joyent, Inc.
17 #
```

```
19 [DEFAULT]
20 pre =
21 verbose = False
22 quiet = False
23 timeout = 60
24 post =
25 outputdir = /var/tmp/test_results
```

```
27 [/opt/util-tests/tests/printf_test]
28 [/opt/util-tests/tests/allowed-ips]
29 [/opt/util-tests/tests/set-linkprop]
30 [/opt/util-tests/tests/libsff/libsff]
```

```
32 [/opt/util-tests/tests/xargs_test]
```

```
34 [/opt/util-tests/tests/mergeq/mqt]
35 [/opt/util-tests/tests/mergeq/wqt]
```

```
37 [/opt/util-tests/tests/libnvpair_json]
38 tests = ['json_00_blank', 'json_01_boolean', 'json_02_numbers',
39 'json_03_empty_arrays', 'json_04_number_arrays', 'json_05_strings',
40 'json_06_nested', 'json_07_nested_arrays']
```

```
42 [/opt/util-tests/tests/grep_test]
43 [/opt/util-tests/tests/date_test]
44 [/opt/util-tests/tests/chown_test]
```

```
46 [/opt/util-tests/tests/demangle]
47 tests = ['afl-fast', 'gcc-libstdc++', 'llvm-stdcxxabi']
```

```
49 [/opt/util-tests/tests/ctf]
50 pre = precheck
51 tests = [ 'ctftest' ]
```

new/usr/src/test/util-tests/tests/ctf/Makefile

1

```
*****
3390 Thu Apr 25 02:20:48 2019
new/usr/src/test/util-tests/tests/ctf/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright (c) 2019, Joyent, Inc.
14 #
15 #
16 include $(SRC)/Makefile.master
17 #
18 ROOTOPTPKG = $(ROOT)/opt/util-tests
19 TESTDIR = $(ROOTOPTPKG)/tests/ctf
20 #
21 SCRIPTS = precheck.ksh \
22           cftftest.ksh \
23           cftftest-convert-non-c.ksh \
24           cftftest-convert-no-dwarf.ksh \
25           cftftest-merge-no-ctf.ksh \
26           cftftest.ksh
27 TESTS = test-float.c \
28         test-reference.c \
29         test-int.c \
30         test-array.c \
31         test-enum.c \
32         test-forward.c \
33         test-sou.c \
34         test-function.c \
35         test-merge-static/Makefile.ctftest \
36         test-merge-static/test-a.c \
37         test-merge-static/test-b.c \
38         test-merge-static/test-c.c \
39         test-merge-static/test-d.c \
40         test-merge-static/test-main.c \
41         test-merge-forward/Makefile.ctftest \
42         test-merge-forward/test-impl.c \
43         test-merge-forward/test-merge.c \
44         test-merge-dedup/Makefile.ctftest \
45         test-merge-dedup/test-merge-1.c \
46         test-merge-dedup/test-merge-2.c \
47         test-merge-dedup/test-merge-3.c \
48         test-merge-dedup/test-merge-dedup.c \
49         test-merge-reduction/Makefile.ctftest \
50         test-merge-reduction/mapfile-vers \
51         test-merge-reduction/test-global.c \
52         test-merge-reduction/test-scoped.c
```

new/usr/src/test/util-tests/tests/ctf/Makefile

2

```
53         test-merge-weak/Makefile.ctftest \
54         test-merge-weak/test-merge-weak.c \
55         test-weak.c \
56         Makefile.ctftest.com
57 #
58 MAKEDIRS = test-merge-static \
59           test-merge-forward \
60           test-merge-dedup \
61           test-merge-reduction \
62           test-merge-weak
63 #
64 CHECKS = check-float-32 \
65          check-float-64 \
66          check-int-32 \
67          check-int-64 \
68          check-reference \
69          check-array \
70          check-enum \
71          check-sou-32 \
72          check-sou-64 \
73          check-forward-32 \
74          check-forward-64 \
75          check-function \
76          check-merge-static \
77          check-merge-forward-32 \
78          check-merge-forward-64 \
79          check-merge-dedup \
80          check-merge-reduction \
81          check-merge-weak \
82          check-weak
83 #
84 COMMON_OBJS = check-common.o
85 ALL_OBJS = $(CHECKS:%=%.o) $(CHECKS:%-32=%.32.o) $(CHECKS:%-64=%.64.o) $(CO
86 #
87 ROOTTESTS = $(TESTS:%=$(TESTDIR)/%)
88 ROOTMAKEDIRS = $(MAKEDIRS:%=$(TESTDIR)/%)
89 ROOTCHECKS = $(CHECKS:%=$(TESTDIR)/%)
90 ROOTSCRIPTS = $(SCRIPTS:%.ksh=$(TESTDIR)/%)
91 #
92 ROOTTESTS := FILEMODE = 0444
93 ROOTCHECKS := FILEMODE = 0555
94 ROOTSCRIPTS := FILEMODE = 0555
95 #
96 include $(SRC)/cmd/Makefile.cmd
97 include $(SRC)/test/Makefile.com
98 #
99 LDLIBS += -lctf
100 #
101 check-merge-static := LDLIBS += -lelf
102 #
103 all: $(CHECKS)
104 #
105 install: all $(ROOTTESTS) $(ROOTCHECKS) $(ROOTSCRIPTS)
106 #
107 $(CHECKS): $(COMMON_OBJS)
108 #
109 clean:
110     $(RM) $(ALL_OBJS)
111 #
112 clobber: clean
113     $(RM) $(CHECKS)
114 #
115 $(ROOTTESTS): $(TESTDIR) $(ROOTMAKEDIRS) $(TESTS)
116 $(ROOTCHECKS): $(TESTDIR) $(CHECKS)
117 $(ROOTSCRIPTS): $(TESTDIR) $(SCRIPTS)
```

new/usr/src/test/util-tests/tests/ctf/Makefile

3

```
119 $(TESTDIR):
120     $(INS.dir)

122 $(ROOTMAKEDIRS):
123     $(INS.dir)

125 $(TESTDIR)/%: %
126     $(INS.file)

128 $(TESTDIR)/%: %.ksh
129     $(INS.rename)

131 %.o: %.c
132     $(COMPILE.c) -o $@ $<
133     $(POST_PROCESS_O)

135 %.32.o: %.c
136     $(COMPILE.c) -o $@ $<
137     $(POST_PROCESS_O)

139 %.64.o: %.c
140     $(COMPILE.c) -DTARGET_LP64 -o $@ $<
141     $(POST_PROCESS_O)

143 %-32: %.32.o
144     $(LINK.c) -o $@ $< $(COMMON_OBJS) $(LDLIBS)
145     $(POST_PROCESS)

147 %-64: %.64.o
148     $(LINK.c) -o $@ $< $(COMMON_OBJS) $(LDLIBS)
149     $(POST_PROCESS)

151 %: %.o
152     $(LINK.c) -o $@ $< $(COMMON_OBJS) $(LDLIBS)
153     $(POST_PROCESS)
```

```

*****
3034 Thu Apr 25 02:20:48 2019
new/usr/src/test/util-tests/tests/ctf/ctftest-convert-no-dwarf.ksh
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
1 #!/usr/bin/ksh
2 #
3 # This file and its contents are supplied under the terms of the
4 # Common Development and Distribution License ("CDDL"), version 1.0.
5 # You may only use this file in accordance with the terms of version
6 # 1.0 of the CDDL.
7 #
8 # A full copy of the text of the CDDL should have accompanied this
9 # source. A copy of the CDDL is also available via the Internet at
10 # http://www.illumos.org/license/CDDL.
11 #
12 # Copyright (c) 2019, Joyent, Inc.
13 #
14
15 set -e
16
17 result=0
18
19 progname=$(basename $0)
20
21 fail()
22 {
23     echo "Failed: $* 2>&1"
24     result=1
25 }
26
27 fail_no_debug()
28 {
29     cmd="$@"
30     set +e
31     out=$(ctf_convert $cmd 2>&1)
32
33     if [[ $? -eq 0 ]]; then
34         fail "$cmd succeeded but should have failed"
35         set -e
36         return;
37     fi
38
39     set -e
40
41     if echo "$out" | grep "is missing debug info" >/dev/null; then
42         return;
43     fi
44
45     if echo "$out" | grep "does not contain DWARF data" >/dev/null; then
46         return;
47     fi
48     fail "$cmd: incorrect output $out"
49 }
50
51 has_ctf()
52 {
53     for f in "$@"; do

```

```

54         if ! elfdump -c -N .SUNW_ctf "$f" |
55             grep '.SUNW_ctf' >/dev/null; then
56             fail "$f lacks CTF section"
57             return
58         fi
59     done
60 }
61
62 cat <<EOF >file1.c
63 #include <stdio.h>
64 struct foo { int a; };
65 int main(void) { struct foo foo = { 4 }; printf("%d\n", foo.a); }
66 EOF
67
68 cat <<EOF >file2.c
69 #include <stdio.h>
70 char myfunc(int a) { printf("%d\n", a); }
71 EOF
72
73 cat <<EOF >file3.c
74 struct bar { char *tar; };
75 void mycxxfunc(char *c) { c[0] = '9'; };
76 EOF
77
78 cat <<EOF >file4.s
79 .globl caller
80 .type caller,@function
81 caller:
82     movl 4(%ebp), %eax
83     ret
84 EOF
85
86 echo "$progname: An empty file should fail conversion due to no DWARF"
87 echo >emptyfile.c
88
89 $ctf_cc -c -o emptyfile.o emptyfile.c
90 fail_no_debug emptyfile.o
91 $ctf_cc -c -o emptyfile.o emptyfile.c
92 $ctf_convert -m emptyfile.o
93
94 $ctf_cc $ctf_debugflags -c -o emptyfile.o emptyfile.c
95 fail_no_debug emptyfile.o
96 $ctf_cc $ctf_debugflags -c -o emptyfile.o emptyfile.c
97 $ctf_convert -m emptyfile.o
98
99 echo "$progname: A file missing DWARF should fail conversion"
100
101 $ctf_cc -c -o file1.o file1.c
102 fail_no_debug file1.o
103 $ctf_cc -c -o file1.o file1.c
104 $ctf_convert -m file1.o
105
106 echo "$progname: A binary with DWARF but 0 debug dies should fail conversion"
107
108 $ctf_cc -o mybin file1.c
109 fail_no_debug mybin
110 $ctf_cc -o mybin file1.c
111 $ctf_convert -m mybin
112
113 echo "$progname: One C file missing DWARF should fail ctfconvert"
114
115 $ctf_cc -c -o file1.o file1.c
116 $ctf_cc $ctf_debugflags -c -o file2.o file2.c
117 ld -r -o files.o file2.o file1.o
118 fail_no_debug files.o
119 ld -r -o files.o file2.o file1.o

```

```
120 $ctf_convert -m files.o
121 has_ctf files.o

123 echo "$progname: One .cc file missing DWARF should pass"

125 $ctf_cc $ctf_debugflags -c -o file1.o file1.c
126 $ctf_cc $ctf_debugflags -c -o file2.o file2.c
127 $ctf_cxx -c -o file3.o file3.cc
128 ld -r -o files.o file1.o file2.o file3.o
129 $ctf_convert files.o
130 has_ctf files.o

132 echo "$progname: One .s file missing DWARF should pass"
133 $ctf_cc $ctf_debugflags -c -o file1.o file1.c
134 $ctf_cc $ctf_debugflags -c -o file2.o file2.c
135 $ctf_as -o file4.o file4.s
136 $ctf_cc -o mybin file1.o file2.o file4.o
137 $ctf_convert mybin
138 has_ctf mybin

140 echo "result is $result"
141 exit $result
```

```

*****
1861 Thu Apr 25 02:20:48 2019
new/usr/src/test/util-tests/tests/ctf/ctftest-convert-non-c.ksh
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
1 #!/usr/bin/ksh
2 #
3 # This file and its contents are supplied under the terms of the
4 # Common Development and Distribution License ("CDDL"), version 1.0.
5 # You may only use this file in accordance with the terms of version
6 # 1.0 of the CDDL.
7 #
8 # A full copy of the text of the CDDL should have accompanied this
9 # source. A copy of the CDDL is also available via the Internet at
10 # http://www.illumos.org/license/CDDL.
11 #
12 # Copyright (c) 2019, Joyent, Inc.
13 #

15 set -e

17 result=0

19 progname=$(basename $0)

21 fail()
22 {
23     echo "Failed: $*" 2>&1
24     result=1
25 }

27 fail_non_c()
28 {
29     cmd="$@"
30     set +e
31     out=$(ctf_convert $cmd 2>&1)

33     if [[ $? -eq 0 ]]; then
34         fail "$cmd succeeded but should have failed"
35         set -e
36         return;
37     fi

39     set -e

41     if ! echo "$out" | \
42         grep "No C source to convert from" >/dev/null; then
43         fail "$cmd: incorrect output $out"
44         return;
45     fi
46 }

48 no_ctf()
49 {
50     for f in "$@"; do
51         if elfdump -c -N .SUNW_ctf "$f" |
52             grep '.SUNW_ctf' >/dev/null; then
53             fail "$f has CTF section"

```

```

54         return
55     fi
56     done
57 }

59 cat <<EOF >file1.c
60 #include <stdio.h>
61 struct foo { int a; };
62 int main(void) { struct foo foo = { 4 }; printf("%d\n", foo.a); }
63 EOF

65 cat <<EOF >file2.cc
66 struct bar { char *tar; };
67 void mycxxfunc(char *c) { c[0] = '9'; };
68 EOF

70 cat <<EOF >file3.s
71 .globl caller
72 .type caller,@function
73 caller:
74     movl 4(%ebp), %eax
75     ret
76 EOF

78 echo "$progname: ctfconvert should fail on a .cc-derived object"
79 $ctf_cxx -c -o file2.o file2.cc
80 fail_non_c file2.o
81 $ctf_cxx -c -o file2.o file2.cc
82 $ctf_convert -i file2.o

84 echo "$progname: ctfconvert shouldn't process .cc-derived DWARF"
85 $ctf_cc $DEBUGFLAGS -c -o file2.o file2.cc
86 $ctf_convert -i file2.o
87 no_ctf file2.o

89 echo "$progname: ctfconvert should fail on a .s-derived object"
90 $ctf_as -o file3.o file3.s
91 fail_non_c file3.o
92 $ctf_as -o file3.o file3.s
93 $ctf_convert -i file3.o
94 no_ctf file3.o

96 echo "result is $result"
97 exit $result

```

```

*****
2489 Thu Apr 25 02:20:48 2019
new/usr/src/test/util-tests/tests/ctf/ctftest-merge-no-ctf.ksh
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
1 #!/usr/bin/ksh
2 #
3 # This file and its contents are supplied under the terms of the
4 # Common Development and Distribution License ("CDDL"), version 1.0.
5 # You may only use this file in accordance with the terms of version
6 # 1.0 of the CDDL.
7 #
8 # A full copy of the text of the CDDL should have accompanied this
9 # source. A copy of the CDDL is also available via the Internet at
10 # http://www.illumos.org/license/CDDL.
11 #
12 # Copyright (c) 2019, Joyent, Inc.
13 #
14
15 set -e
16
17 result=0
18
19 progname=$(basename $0)
20
21 fail()
22 {
23     echo "Failed: $" 2>&1
24     result=1
25 }
26
27 fail_no_ctf()
28 {
29     cmd="$@"
30     set +e
31     out=$(($cmd 2>&1)
32
33     if [[ $? -eq 0 ]]; then
34         fail "$cmd succeeded but should have failed"
35     else
36         return;
37     fi
38
39     set -e
40
41     if ! echo "$out" | \
42         grep "File does not contain CTF data" >/dev/null; then
43         fail "$cmd: incorrect output $out"
44     else
45         return;
46     fi
47 }
48
49 has_ctf()
50 {
51     for f in "$@"; do
52         if ! elfdump -c -N .SUNW_ctf "$f" |
53             grep '.SUNW_ctf' >/dev/null; then
54             fail "$f lacks CTF section"
55         fi
56     done
57 }

```

```

54     return
55     fi
56     done
57 }
58
59 cat <<EOF >file1.c
60 #include <stdio.h>
61 struct foo { int a; };
62 struct foo foos[400];
63 int main(void) { struct foo foo = { 4 }; printf("%d\n", foo.a); }
64 EOF
65
66 cat <<EOF >file2.c
67 #include <stdio.h>
68 struct foo { char b; float c; };
69 struct foo stuff[90];
70 char myfunc(int a) { printf("%d\n", a); }
71 EOF
72
73 cat <<EOF >file3.c
74 struct bar { char *tar; };
75 void mycxxfunc(char *c) { c[0] = '9'; };
76 EOF
77
78 cat <<EOF >file4.s
79 .globl caller
80 .type caller,@function
81 caller:
82     movl 4(%ebp), %eax
83     ret
84 EOF
85
86 echo "$progname: ctfmerge should fail if one C-source lacks CTF"
87
88 $ctf_cc $ctf_debugflags -c -o file1.o file1.c
89 $ctf_convert file1.o
90 $ctf_cc -c -o file2.o file2.c
91 ld -r -o files.o file2.o file1.o
92 fail_no_ctf $ctf_merge -o files.o file2.o file1.o
93 ld -r -o files.o file2.o file1.o
94 $ctf_merge -m -o files.o file2.o file1.o
95 has_ctf files.o
96 $ctf_cc -o mybin file2.o file1.o
97 fail_no_ctf $ctf_merge -o mybin file2.o file1.o
98 $ctf_cc -o mybin file2.o file1.o
99 $ctf_merge -m -o mybin file2.o file1.o
100
101
102 echo "$progname: ctfmerge should allow a .cc file to lack CTF"
103 $ctf_cxx -c -o file3.o file3.c
104 ld -r -o files.o file1.o file3.o
105 $ctf_merge -o files.o file1.o file3.o
106 ld -r -o files.o file1.o file3.o
107 $ctf_merge -m -o files.o file1.o file3.o
108
109 echo "$progname: ctfmerge should allow an .s file to lack CTF"
110 $ctf_as -o file4.o file4.s
111 ld -r -o files.o file4.o file1.o
112 $ctf_merge -o files.o file4.o file1.o
113 ld -r -o files.o file4.o file1.o
114 $ctf_merge -m -o files.o file4.o file1.o
115
116 echo "result is $result"
117 exit $result

```


new/usr/src/test/util-tests/tests/ctf/ctftest.ksh

1

```
*****
5845 Thu Apr 25 02:20:48 2019
new/usr/src/test/util-tests/tests/ctf/ctftest.ksh
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
1 #!/usr/bin/ksh
2 #
3 #
4 # This file and its contents are supplied under the terms of the
5 # Common Development and Distribution License ("CDDL"), version 1.0.
6 # You may only use this file in accordance with the terms of version
7 # 1.0 of the CDDL.
8 #
9 # A full copy of the text of the CDDL should have accompanied this
10 # source. A copy of the CDDL is also available via the Internet at
11 # http://www.illumos.org/license/CDDL.
12 #
13 #
14 #
15 # Copyright (c) 2019, Joyent, Inc.
16 #
17 #
18 #
19 # Run all of the various CTF tests
20 #
21 #
22 unalias -a
23 #set -o xtrace
24 #
25 if [[ -z "$TMPDIR" ]]; then
26     TMPDIR="/tmp"
27 fi
28 #
29 #
30 ctf_arg0=$(basename $0)
31 ctf_root=$(cd $(dirname $0) && echo $PWD)
32 ctf_tests=
33 ctf_cc="gcc"
34 ctf_cxx="g++"
35 ctf_as="as"
36 ctf_compiler="gcc"
37 ctf_convert="ctfconvert"
38 ctf_merge="ctfmerge"
39 ctf_debugflags="-gdwarf-2 "
40 ctf_mach32="-m32"
41 ctf_mach64="-m64"
42 ctf_32cflags="$ctf_mach32 $ctf_debugflags"
43 ctf_64cflags="$ctf_mach64 $ctf_debugflags"
44 ctf_temp="$TMPDIR/ctftest.$$"
45 ctf_makefile="Makefile.ctftest"
46 ctf_nerrs=0
47 #
48 usage()
49 {
50     typeset msg="$*"
51     [[ -z "$msg" ]] || echo "$msg" >&2
52     cat <<USAGE >&2
53 Usage: $ctf_arg0 [-a as] [-c cc] [-C CC] [-g flags] [-m ctfmerge] [-t ctfconvert]
```

new/usr/src/test/util-tests/tests/ctf/ctftest.ksh

2

```
49 Usage: $ctf_arg0 [-c compiler] [-g flags] [-m ctfmerge] [-t ctfconvert]
50 #
51 #
52 Runs the CTF test suite
53 #
54 -a assembler Use the specified assembler, defaults to 'as'
55 -c compiler Use the specified C compiler, defaults to 'gcc'
56 -C compiler Use the specified C++ compiler, defaults to 'g++'
57 -c compiler Use the specified compiler, defaults to 'gcc'
58 on path.
59 -g flags Use flags to generate debug info. Defaults to
60 "-gdwarf-2".
61 -m ctfmerge Use the specified ctfmerge, defaults to
62 'ctfmerge'
63 'ctfmerge' on path.
64 -t ctfconvert Use the specified ctfconvert, defaults to
65 'ctfconvert'
66 'ctfconvert' on path.
67 #
68 USAGE
69 exit 2
70 }
71 #
72 }
73 #
74 }
75 #
76 }
77 #
78 }
79 #
80 }
81 #
82 }
83 check_env()
84 {
85     if which "$1" 2>/dev/null >/dev/null; then
86         return
87     fi
88     [[ -f "$1" ]] || fatal "failed to find tool $1"
89 }
90 #
91 #
92 #
93 #
94 #
95 #
96 #
97 #
98 #
99 #
100 #
101 #
102 #
103 #
104 #
105 #
106 #
107 #
108 #
109 #
110 #
111 #
112 #
113 #
114 #
115 #
116 #
117 #
118 #
119 #
120 #
121 #
122 #
123 #
124 #
125 #
126 #
127 #
128 #
129 #
130 #
131 #
132 #
133 #
134 #
135 #
136 #
137 #
138 #
139 #
140 #
141 #
142 #
143 #
144 #
145 #
146 #
147 #
148 #
149 #
150 #
151 #
152 #
153 #
154 #
155 #
156 #
157 #
158 #
159 #
160 #
161 #
162 #
163 #
164 #
165 #
166 #
167 #
168 #
169 #
170 #
171 #
172 #
173 #
174 #
175 #
176 #
177 #
178 #
179 #
180 #
181 #
182 #
183 #
184 #
185 #
186 #
187 #
188 #
189 #
190 #
191 #
192 #
193 #
194 #
195 #
196 #
197 #
198 #
199 #
200 #
201 #
202 #
203 #
204 #
205 #
206 #
207 #
208 #
209 #
210 #
211 #
212 #
213 #
214 #
215 #
216 #
217 #
218 #
219 #
220 #
221 #
222 #
223 #
224 #
225 #
226 #
227 #
228 #
229 #
230 #
231 #
232 #
233 #
234 #
235 #
236 #
237 #
238 #
239 #
240 #
241 #
242 #
243 #
244 #
245 #
246 #
247 #
248 #
249 #
250 #
251 #
252 #
253 #
254 #
255 #
256 #
257 #
258 #
259 #
260 #
261 #
262 #
263 #
264 #
265 #
266 #
267 #
268 #
269 #
270 #
271 #
272 #
273 #
274 #
275 #
276 #
277 #
278 #
279 #
280 #
281 #
282 #
283 #
284 #
285 #
286 #
287 #
288 #
289 #
290 #
291 #
292 #
293 #
294 #
295 #
296 #
297 #
298 #
299 #
300 #
301 #
302 #
303 #
304 #
305 #
306 #
307 #
308 #
309 #
310 #
311 #
312 #
313 #
314 #
315 #
316 #
317 #
318 #
319 #
320 #
321 #
322 #
323 #
324 #
325 #
326 #
327 #
328 #
329 #
330 #
331 #
332 #
333 #
334 #
335 #
336 #
337 #
338 #
339 #
340 #
341 #
342 #
343 #
344 #
345 #
346 #
347 #
348 #
349 #
350 #
351 #
352 #
353 #
354 #
355 #
356 #
357 #
358 #
359 #
360 #
361 #
362 #
363 #
364 #
365 #
366 #
367 #
368 #
369 #
370 #
371 #
372 #
373 #
374 #
375 #
376 #
377 #
378 #
379 #
380 #
381 #
382 #
383 #
384 #
385 #
386 #
387 #
388 #
389 #
390 #
391 #
392 #
393 #
394 #
395 #
396 #
397 #
398 #
399 #
400 #
401 #
402 #
403 #
404 #
405 #
406 #
407 #
408 #
409 #
410 #
411 #
412 #
413 #
414 #
415 #
416 #
417 #
418 #
419 #
420 #
421 #
422 #
423 #
424 #
425 #
426 #
427 #
428 #
429 #
430 #
431 #
432 #
433 #
434 #
435 #
436 #
437 #
438 #
439 #
440 #
441 #
442 #
443 #
444 #
445 #
446 #
447 #
448 #
449 #
450 #
451 #
452 #
453 #
454 #
455 #
456 #
457 #
458 #
459 #
460 #
461 #
462 #
463 #
464 #
465 #
466 #
467 #
468 #
469 #
470 #
471 #
472 #
473 #
474 #
475 #
476 #
477 #
478 #
479 #
480 #
481 #
482 #
483 #
484 #
485 #
486 #
487 #
488 #
489 #
490 #
491 #
492 #
493 #
494 #
495 #
496 #
497 #
498 #
499 #
500 #
501 #
502 #
503 #
504 #
505 #
506 #
507 #
508 #
509 #
510 #
511 #
512 #
513 #
514 #
515 #
516 #
517 #
518 #
519 #
520 #
521 #
522 #
523 #
524 #
525 #
526 #
527 #
528 #
529 #
530 #
531 #
532 #
533 #
534 #
535 #
536 #
537 #
538 #
539 #
540 #
541 #
542 #
543 #
544 #
545 #
546 #
547 #
548 #
549 #
550 #
551 #
552 #
553 #
554 #
555 #
556 #
557 #
558 #
559 #
560 #
561 #
562 #
563 #
564 #
565 #
566 #
567 #
568 #
569 #
570 #
571 #
572 #
573 #
574 #
575 #
576 #
577 #
578 #
579 #
580 #
581 #
582 #
583 #
584 #
585 #
586 #
587 #
588 #
589 #
590 #
591 #
592 #
593 #
594 #
595 #
596 #
597 #
598 #
599 #
600 #
601 #
602 #
603 #
604 #
605 #
606 #
607 #
608 #
609 #
610 #
611 #
612 #
613 #
614 #
615 #
616 #
617 #
618 #
619 #
620 #
621 #
622 #
623 #
624 #
625 #
626 #
627 #
628 #
629 #
630 #
631 #
632 #
633 #
634 #
635 #
636 #
637 #
638 #
639 #
640 #
641 #
642 #
643 #
644 #
645 #
646 #
647 #
648 #
649 #
650 #
651 #
652 #
653 #
654 #
655 #
656 #
657 #
658 #
659 #
660 #
661 #
662 #
663 #
664 #
665 #
666 #
667 #
668 #
669 #
670 #
671 #
672 #
673 #
674 #
675 #
676 #
677 #
678 #
679 #
680 #
681 #
682 #
683 #
684 #
685 #
686 #
687 #
688 #
689 #
690 #
691 #
692 #
693 #
694 #
695 #
696 #
697 #
698 #
699 #
700 #
701 #
702 #
703 #
704 #
705 #
706 #
707 #
708 #
709 #
710 #
711 #
712 #
713 #
714 #
715 #
716 #
717 #
718 #
719 #
720 #
721 #
722 #
723 #
724 #
725 #
726 #
727 #
728 #
729 #
730 #
731 #
732 #
733 #
734 #
735 #
736 #
737 #
738 #
739 #
740 #
741 #
742 #
743 #
744 #
745 #
746 #
747 #
748 #
749 #
750 #
751 #
752 #
753 #
754 #
755 #
756 #
757 #
758 #
759 #
760 #
761 #
762 #
763 #
764 #
765 #
766 #
767 #
768 #
769 #
770 #
771 #
772 #
773 #
774 #
775 #
776 #
777 #
778 #
779 #
780 #
781 #
782 #
783 #
784 #
785 #
786 #
787 #
788 #
789 #
790 #
791 #
792 #
793 #
794 #
795 #
796 #
797 #
798 #
799 #
800 #
801 #
802 #
803 #
804 #
805 #
806 #
807 #
808 #
809 #
810 #
811 #
812 #
813 #
814 #
815 #
816 #
817 #
818 #
819 #
820 #
821 #
822 #
823 #
824 #
825 #
826 #
827 #
828 #
829 #
830 #
831 #
832 #
833 #
834 #
835 #
836 #
837 #
838 #
839 #
840 #
841 #
842 #
843 #
844 #
845 #
846 #
847 #
848 #
849 #
850 #
851 #
852 #
853 #
854 #
855 #
856 #
857 #
858 #
859 #
860 #
861 #
862 #
863 #
864 #
865 #
866 #
867 #
868 #
869 #
870 #
871 #
872 #
873 #
874 #
875 #
876 #
877 #
878 #
879 #
880 #
881 #
882 #
883 #
884 #
885 #
886 #
887 #
888 #
889 #
890 #
891 #
892 #
893 #
894 #
895 #
896 #
897 #
898 #
899 #
900 #
901 #
902 #
903 #
904 #
905 #
906 #
907 #
908 #
909 #
910 #
911 #
912 #
913 #
914 #
915 #
916 #
917 #
918 #
919 #
920 #
921 #
922 #
923 #
924 #
925 #
926 #
927 #
928 #
929 #
930 #
931 #
932 #
933 #
934 #
935 #
936 #
937 #
938 #
939 #
940 #
941 #
942 #
943 #
944 #
945 #
946 #
947 #
948 #
949 #
950 #
951 #
952 #
953 #
954 #
955 #
956 #
957 #
958 #
959 #
960 #
961 #
962 #
963 #
964 #
965 #
966 #
967 #
968 #
969 #
970 #
971 #
972 #
973 #
974 #
975 #
976 #
977 #
978 #
979 #
980 #
981 #
982 #
983 #
984 #
985 #
986 #
987 #
988 #
989 #
990 #
991 #
992 #
993 #
994 #
995 #
996 #
997 #
998 #
999 #
1000 #
1001 #
1002 #
1003 #
1004 #
1005 #
1006 #
1007 #
1008 #
1009 #
1010 #
1011 #
1012 #
1013 #
1014 #
1015 #
1016 #
1017 #
1018 #
1019 #
1020 #
1021 #
1022 #
1023 #
1024 #
1025 #
1026 #
1027 #
1028 #
1029 #
1030 #
1031 #
1032 #
1033 #
1034 #
1035 #
1036 #
1037 #
1038 #
1039 #
1040 #
1041 #
1042 #
1043 #
1044 #
1045 #
1046 #
1047 #
1048 #
1049 #
1050 #
1051 #
1052 #
1053 #
1054 #
1055 #
1056 #
1057 #
1058 #
1059 #
1060 #
1061 #
1062 #
1063 #
1064 #
1065 #
1066 #
1067 #
1068 #
1069 #
1070 #
1071 #
1072 #
1073 #
1074 #
1075 #
1076 #
1077 #
1078 #
1079 #
1080 #
1081 #
1082 #
1083 #
1084 #
1085 #
1086 #
1087 #
1088 #
1089 #
1090 #
1091 #
1092 #
1093 #
1094 #
1095 #
1096 #
1097 #
1098 #
1099 #
1100 #
1101 #
1102 #
1103 #
1104 #
1105 #
1106 #
1107 #
1108 #
1109 #
1110 #
1111 #
1112 #
1113 #
1114 #
1115 #
1116 #
1117 #
1118 #
1119 #
1120 #
1121 #
1122 #
1123 #
1124 #
1125 #
1126 #
1127 #
1128 #
1129 #
1130 #
1131 #
1132 #
1133 #
1134 #
1135 #
1136 #
1137 #
1138 #
1139 #
1140 #
1141 #
1142 #
1143 #
1144 #
1145 #
1146 #
1147 #
1148 #
1149 #
1150 #
1151 #
1152 #
1153 #
1154 #
1155 #
1156 #
1157 #
1158 #
1159 #
1160 #
1161 #
1162 #
1163 #
1164 #
1165 #
1166 #
1167 #
1168 #
1169 #
1170 #
1171 #
1172 #
1173 #
1174 #
1175 #
1176 #
1177 #
1178 #
1179 #
1180 #
1181 #
1182 #
1183 #
1184 #
1185 #
1186 #
1187 #
1188 #
1189 #
1190 #
1191 #
1192 #
1193 #
1194 #
1195 #
1196 #
1197 #
1198 #
1199 #
1200 #
1201 #
1202 #
1203 #
1204 #
1205 #
1206 #
1207 #
1208 #
1209 #
1210 #
1211 #
1212 #
1213 #
1214 #
1215 #
1216 #
1217 #
1218 #
1219 #
1220 #
1221 #
1222 #
1223 #
1224 #
1225 #
1226 #
1227 #
1228 #
1229 #
1230 #
1231 #
1232 #
1233 #
1234 #
1235 #
1236 #
1237 #
1238 #
1239 #
1240 #
1241 #
1242 #
1243 #
1244 #
1245 #
1246 #
1247 #
1248 #
1249 #
1250 #
1251 #
1252 #
1253 #
1254 #
1255 #
1256 #
1257 #
1258 #
1259 #
1260 #
1261 #
1262 #
1263 #
1264 #
1265 #
1266 #
1267 #
1268 #
1269 #
1270 #
1271 #
1272 #
1273 #
1274 #
1275 #
1276 #
1277 #
1278 #
1279 #
1280 #
1281 #
1282 #
1283 #
1284 #
1285 #
1286 #
1287 #
1288 #
1289 #
1290 #
1291 #
1292 #
1293 #
1294 #
1295 #
1296 #
1297 #
1298 #
1299 #
1300 #
1301 #
1302 #
1303 #
1304 #
1305 #
1306 #
1307 #
1308 #
1309 #
1310 #
1311 #
1312 #
1313 #
1314 #
1315 #
1316 #
1317 #
1318 #
1319 #
1320 #
1321 #
1322 #
1323 #
1324 #
1325 #
1326 #
1327 #
1328 #
1329 #
1330 #
1331 #
1332 #
1333 #
1334 #
1335 #
1336 #
1337 #
1338 #
1339 #
1340 #
1341 #
1342 #
1343 #
1344 #
1345 #
1346 #
1347 #
1348 #
1349 #
1350 #
1351 #
1352 #
1353 #
1354 #
1355 #
1356 #
1357 #
1358 #
1359 #
1360 #
1361 #
1362 #
1363 #
1364 #
1365 #
1366 #
1367 #
1368 #
1369 #
1370 #
1371 #
1372 #
1373 #
1374 #
1375 #
1376 #
1377 #
1378 #
1379 #
1380 #
1381 #
1382 #
1383 #
1384 #
1385 #
1386 #
1387 #
1388 #
1389 #
1390 #
1391 #
1392 #
1393 #
1394 #
1395 #
1396 #
1397 #
1398 #
1399 #
1400 #
1401 #
1402 #
1403 #
1404 #
1405 #
1406 #
1407 #
1408 #
1409 #
1410 #
1411 #
1412 #
1413 #
1414 #
1415 #
1416 #
1417 #
1418 #
1419 #
1420 #
1421 #
1422 #
1423 #
1424 #
1425 #
1426 #
1427 #
1428 #
1429 #
1430 #
1431 #
1432 #
1433 #
1434 #
1435 #
1436 #
1437 #
1438 #
1439 #
1440 #
1441 #
1442 #
1443 #
1444 #
1445 #
1446 #
1447 #
1448 #
1449 #
1450 #
1451 #
1452 #
1453 #
1454 #
1455 #
1456 #
1457 #
1458 #
1459 #
1460 #
1461 #
1462 #
1463 #
1464 #
1465 #
1466 #
1467 #
1468 #
1469 #
1470 #
1471 #
1472 #
1473 #
1474 #
1475 #
1476 #
1477 #
1478 #
1479 #
1480 #
1481 #
1482 #
1483 #
1484 #
1485 #
1486 #
1487 #
1488 #
1489 #
1490 #
1491 #
1492 #
1493 #
1494 #
1495 #
1496 #
1497 #
1498 #
1499 #
1500 #
1501 #
1502 #
1503 #
1504 #
1505 #
1506 #
1507 #
1508 #
1509 #
1510 #
1511 #
1512 #
1513 #
1514 #
1515 #
1516 #
1517 #
1518 #
1519 #
1520 #
1521 #
1522 #
1523 #
1524 #
1525 #
1526 #
1527 #
1528 #
1529 #
1530 #
1531 #
1532 #
1533 #
1534 #
1535 #
1536 #
1537 #
1538 #
1539 #
1540 #
1541 #
1542 #
1543 #
1544 #
1545 #
1546 #
1547 #
1548 #
1549 #
1550 #
1551 #
1552 #
1553 #
1554 #
1555 #
1556 #
1557 #
1558 #
1559 #
1560 #
1561 #
1562 #
1563 #
1564 #
1565 #
1566 #
1567 #
1568 #
1569 #
1570 #
1571 #
1572 #
1573 #
1574 #
1575 #
1576 #
1577 #
1578 #
1579 #
1580 #
1581 #
1582 #
1583 #
1584 #
1585 #
1586 #
1587 #
1588 #
1589 #
1590 #
1591 #
1592 #
1593 #
1594 #
1595 #
1596 #
1597 #
1598 #
1599 #
1600 #
1601 #
1602 #
1603 #
1604 #
1605 #
1606 #
1607 #
1608 #
1609 #
1610 #
1611 #
1612 #
1613 #
1614 #
1615 #
1616 #
1617 #
1618 #
1619 #
1620 #
1621 #
1622 #
1623 #
1624 #
1625 #
1626 #
1627 #
1628 #
1629 #
1630 #
1631 #
1632 #
1633 #
1634 #
1635 #
1636 #
1637 #
1638 #
1639 #
1640 #
1641 #
1642 #
1643 #
1644 #
1645 #
1646 #
1647 #
1648 #
1649 #
1650 #
1651 #
1652 #
1653 #
1654 #
1655 #
1656 #
1657 #
1658 #
1659 #
1660 #
1661 #
1662 #
1663 #
1664 #
1665 #
1666 #
1667 #
1668 #
1669 #
1670 #
1671 #
1672 #
1673 #
1674 #
1675 #
1676 #
1677 #
1678 #
1679 #
1680 #
1681 #
1682 #
1683 #
1684 #
1685 #
1686 #
1687 #
1688 #
1689 #
1690 #
1691 #
1692 #
1693 #
1694 #
1695 #
1696 #
1697 #
1698 #
1699 #
1700 #
1701 #
1702 #
1703 #
1704 #
1705 #
1706 #
1707 #
1708 #
1709 #
1710 #
1711 #
1712 #
1713 #
1714 #
1715 #
1716 #
1717 #
1718 #
1719 #
1720 #
1721 #
1722 #
1723 #
1724 #
1725 #
1726 #
1727 #
1728 #
1729 #
1730 #
1731 #
1732 #
1733 #
1734 #
1735 #
1736 #
1737 #
1738 #
1739 #
1740 #
1741 #
1742 #
1743 #
1744 #
1745 #
1746 #
1747 #
1748 #
1749 #
1750 #
1751 #
1752 #
1753 #
1754 #
1755 #
1756 #
1757 #
1758 #
1759 #
1760 #
1761 #
1762 #
1763 #
1764 #
1765 #
1766 #
1767 #
1768 #
1769 #
1770 #
1771 #
1772 #
1773 #
1774 #
1775 #
1776 #
1777 #
1778 #
1779 #
1780 #
1781 #
1782 #
1783 #
1784 #
1785 #
1786 #
1787 #
1788 #
1789 #
1790 #
1791 #
1792 #
1793 #
1794 #
1795 #
1796 #
1797 #
1798 #
1799 #
1800 #
1801 #
1802 #
1803 #
1804 #
1805 #
1806 #
1807 #
1808 #
1809 #
1810 #
1811 #
1812 #
1813 #
1814 #
1815 #
1816 #
1817 #
1818 #
1819 #
1820 #
1821 #
1822 #
1823 #
1824 #
1825 #
1826 #
1827 #
1828 #
1829 #
1830 #
1831 #
1832 #
1833 #
1834 #
1835 #
1836 #
1837 #
1838 #
1839 #
1840 #
1841 #
1842 #
1843 #
1844 #
1845 #
1846 #
1847 #
1848 #
1849 #
1850 #
1851 #
1852 #
1853 #
1854 #
1855 #
1856 #
1857 #
1858 #
1859 #
1860 #
1861 #
1862 #
1863 #
1864 #
1865 #
1866 #
1867 #
1868 #
1869 #
1870 #
1871 #
1872 #
1873 #
1874 #
1875 #
1876 #
1877 #
1878 #
1879 #
1880 #
1881 #
1882 #
1883 #
1884 #
1885 #
1886 #
1887 #
1888 #
1889 #
1890 #
1891 #
1892 #
1893 #
1894 #
1895 #
1896 #
1897 #
1898 #
1899 #
1900 #
1901 #
1902 #
1903 #
1904 #
1905 #
1906 #
1907 #
1908 #
1909 #
1910 #
1911 #
1912 #
1913 #
1914 #
1915 #
1916 #
1917 #
1918 #
1919 #
1920 #
1921 #
1922 #
1923 #
1924 #
1925 #
1926 #
1927 #
1928 #
1929 #
1930 #
1931 #
1932 #
1933 #
1934 #
1935 #
1936 #
1937 #
1938 #
1939 #
1940 #
1941 #
1942 #
1943 #
1944 #
1945 #
1946 #
1947 #
1948 #
1949 #
1950 #
1951 #
1952 #
1953 #
1954 #
1955 #
1956 #
1957 #
1958 #
1959 #
1960 #
1961 #
1962 #
1963 #
1964 #
1965 #
1966 #
1967 #
1968 #
1969 #
1970 #
1971 #
1972 #
1973 #
1974 #
1975 #
1976 #
1977 #
1978 #
1979 #
1980 #
1981 #
1982 #
1983 #
1984 #
1985 #
1986 #
1987 #
1988 #
1989 #
1990 #
1991 #
1992 #
1993 #
1994 #
1995 #
1996 #
1997 #
1998 #
1999 #
2000 #
2001 #
2002 #
2003 #
2004 #
2005 #
2006 #
2007 #
2008 #
2009 #
2010 #
2011 #
2012 #
2013 #
2014 #
2015 #
2016 #
2017 #
2018 #
2019 #
2020 #
2021 #
2022 #
2023 #
2024 #
2025 #
2026 #
2027 #
2028 #
2029 #
2030 #
2031 #
2032 #
2033 #
2034 #
2035 #
2036 #
2037 #
2038 #
2039 #
2040 #
2041 #
2042 #
2043 #
2044 #
2045 #
2046 #
2047 #
2048 #
2049 #
2050 #
2051 #
2052 #
2053 #
2054 #
2055 #
2056 #
2057 #
2058 #
2059 #
2060 #
2061 #
2062 #
2063 #
2064 #
2065 #
2066 #
2067 #
2068 #
2069 #
2070 #
2071 #
2072 #
2073 #
2074 #
2075 #
2076 #
2077 #
2078 #
2079 #
2080 #
2081 #
2082 #
2083 #
20
```

```

116         return
117     fi

119     echo "TEST PASSED: $source $flags"
120 }

122 #
123 # Perform a more complex build. The Makefile present will drive the
124 # building of the artifacts and the running of the tests based on the
125 # variables that we pass to it.
126 #
127 run_dir()
128 {
129     typeset dir outdir check32 check64 flags32 flags64

131     dir=$1
132     outdir="$TMPDIR/ctftest.$$-(basename $d)"
133     check32=$2
134     flags32=$3
135     check64=$4
136     flags64=$5

138     if ! mkdir $outdir; then
139         fatal "failed to make temporary directory '$outdir'"
140     fi

142     if ! make -C $dir -f Makefile.ctftest \
143         BUILDDIR="$outdir" \
144         CC="$ctf_cc" \
145         CC="$ctf_compiler" \
146         CFLAGS32="$ctf_mach32" \
147         CFLAGS64="$ctf_mach64" \
148         DEBUGFLAGS="$ctf_debugflags" \
149         CTFCONVERT="$ctf_convert" \
150         CTFMERGE="$ctf_merge" \
151         build 1>/dev/null; then
152         rm -rf $outdir
153         test_fail "failed to build $dir"
154         return
155     fi

156     if ! make -C $dir -f Makefile.ctftest \
157         BUILDDIR="$outdir" \
158         CHECK32="$check32" \
159         CHECK64="$check64" \
160         run-test 1>/dev/null; then
161         rm -rf $outdir
162         test_fail "failed to run tests for $dir"
163         return
164     fi

166     rm -rf $outdir
167     echo "TEST PASSED: $dir (dir)"
168 }

170 #
171 # Find all of the tests that exist and then try to run them all. Tests
172 # may either be a single file or a directory.
173 #
174 run_tests()
175 {
176     typeset t base check
177     ctf_tests=$(ls "$ctf_root"/*.c)
178     for t in $ctf_tests; do
179         base=$(basename "$t" .c)
180         check=$(echo "$base" | sed s/test-/check-/)

```

```

181     if [[ -f "$ctf_root/$check" ]]; then
182         run_one $t "$ctf_root/$check" "$ctf_32cflags"
183         run_one $t "$ctf_root/$check" "$ctf_64cflags"
184     elif [[ -f "$ctf_root/$check-32" && \
185         -f "$ctf_root/$check-64" ]]; then
186         run_one $t "$ctf_root/$check-32" "$ctf_32cflags"
187         run_one $t "$ctf_root/$check-64" "$ctf_64cflags"
188     else
189         test_fail "missing checker for $t"
190     fi
191 done

193 for d in $(find "$ctf_root" -maxdepth 1 -type d -name 'test-*'); do
194     [[ ! -f "$d/$ctf_makefile" ]] && continue
195     base=$(basename "$d")
196     check=$(echo "$base" | sed s/test-/check-/)
197     if [[ -f "$ctf_root/$check" ]]; then
198         run_dir $d "$ctf_root/$check" "$ctf_32cflags" \
199             "$ctf_root/$check" "$ctf_64cflags"
200     elif [[ -f "$ctf_root/$check-32" && \
201         -f "$ctf_root/$check-64" ]]; then
202         run_dir $d "$ctf_root/$check-32" "$ctf_32cflags" \
203             "$ctf_root/$check-64" "$ctf_64cflags"
204     else
205         test_fail "missing checker for $t"
206     fi
207 done

209     outdir="$TMPDIR/ctftest.$$"

211     for f in $(find "$ctf_root" -maxdepth 1 -type f -name 'ctftest-*'); do
212         if ! mkdir $outdir; then
213             fatal "failed to make temporary directory '$outdir'"
214         fi

216         echo "Running $f in $outdir"

218         (cd $outdir && $f)

220         if [[ $? -ne 0 ]]; then
221             test_fail "$f failed"
222         else
223             echo "TEST PASSED: $f"
224         fi

226         rm -rf $outdir
227     done
228 }

230 while getopts ":a:C:c:g:m:t:" c $@; do
231     while getopts ":c:g:m:t:" c $@; do
232         case "$c" in
233             a)
234                 ctf_as=$OPTARG
235                 ;;
236             C)
237                 ctf_cxx=$OPTARG
238                 ;;
239             c)
240                 ctf_cc=$OPTARG
241                 ctf_compiler=$OPTARG
242                 ;;
243             g)
244                 ctf_debugflags=$OPTARG
245                 ;;
246             m)

```

```
245         ctf_merge=$OPTARG
246         ;;
247     t)
248         ctf_convert=$OPTARG
249         ;;
250     :)
251         usage "option requires an argument -- $OPTARG"
252         ;;
253     *)
254         usage "invalid option -- $OPTARG"
255         ;;
256     esac
257 done

259 ctf_32cflags="$ctf_mach32 $ctf_debugflags"
260 ctf_64cflags="$ctf_mach64 $ctf_debugflags"

262 export ctf_as ctf_cc ctf_cxx ctf_debugflags ctf_merge ctf_convert

241 check_env "$ctf_compiler"
242 check_env "$ctf_convert"
243 check_env "$ctf_merge"
264 announce

266 run_tests

268 if [[ $ctf_nerrs -ne 0 ]]; then
269     if [[ $ctf_nerrs -eq 1 ]]; then
270         printf "\n%s: %u test failed\n" "$ctf_arg0" "$ctf_nerrs"
271     else
272         printf "\n%s: %u tests failed\n" "$ctf_arg0" "$ctf_nerrs"
273     fi
274     exit 1
275 else
276     printf "\n%s: All tests passed successfully\n" "$ctf_arg0"
277     exit 0
278 fi
```

new/usr/src/test/util-tests/tests/ctf/precheck.ksh

1

```
*****
737 Thu Apr 25 02:20:49 2019
new/usr/src/test/util-tests/tests/ctf/precheck.ksh
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
```

```
1 #!/usr/bin/ksh
2 #
3 #
4 # This file and its contents are supplied under the terms of the
5 # Common Development and Distribution License ("CDDL"), version 1.0.
6 # You may only use this file in accordance with the terms of version
7 # 1.0 of the CDDL.
8 #
9 # A full copy of the text of the CDDL should have accompanied this
10 # source. A copy of the CDDL is also available via the Internet at
11 # http://www.illumos.org/license/CDDL.
12 #
13 #
14 #
15 # Copyright (c) 2019, Joyent, Inc.
16 #
17 #
18 unalias -a
19 #
20 check_env()
21 {
22     if which "$1" 2>/dev/null >/dev/null; then
23         return
24     fi
25 #
26     [[ -f "$1" ]] || {
27         echo "failed to find $1" >&2
28         exit 1
29     }
30 }
31 #
32 check_env as
33 check_env ctfconvert
34 check_env ctmerge
35 check_env elfdump
36 check_env gcc
37 check_env g++
38 check_env ld
39 check_env make
```

```

*****
74634 Thu Apr 25 02:20:49 2019
new/usr/src/uts/common/Makefile.rules
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 #
23 # Copyright (c) 1991, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright 2016 Garrett D'Amore <garrett@damore.org>
25 # Copyright 2013 Saso Kiselkov. All rights reserved.
26 # Copyright 2019 Joyent, Inc.
26 # Copyright 2016 Joyent, Inc.
27 # Copyright 2018 Nexenta Systems, Inc.
28 # Copyright (c) 2016 by Delphix. All rights reserved.
29 #
30 #
31 #
32 # uts/common/Makefile.rules
33 #
34 # This Makefile defines all the file build rules for the directory
35 # uts/common and its children. These are the source files which may
36 # be considered common to all SunOS systems.
37 #
38 # The following two-level ordering must be maintained in this file.
39 # Lines are sorted first in order of decreasing specificity based on
40 # the first directory component. That is, sun4u rules come before
41 # sparc rules come before common rules.
42 #
43 # Lines whose initial directory components are equal are sorted
44 # alphabetically by the remaining components.
45 #
46 #
47 # Section 1a: C objects build rules
48 #
49 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/aes/%.c
50 $(COMPILE.c) -o $@ $<
51 $(CTFCONVERT_O)

```

```

53 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/arcfour/%.c
54 $(COMPILE.c) -o $@ $<
55 $(CTFCONVERT_O)
56 #
57 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/blowfish/%.c
58 $(COMPILE.c) -o $@ $<
59 $(CTFCONVERT_O)
60 #
61 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/ecc/%.c
62 $(COMPILE.c) -o $@ $<
63 $(CTFCONVERT_O)
64 #
65 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/modes/%.c
66 $(COMPILE.c) -o $@ $<
67 $(CTFCONVERT_O)
68 #
69 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/padding/%.c
70 $(COMPILE.c) -o $@ $<
71 $(CTFCONVERT_O)
72 #
73 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/rng/%.c
74 $(COMPILE.c) -o $@ $<
75 $(CTFCONVERT_O)
76 #
77 $(OBJS_DIR)/%.o: $(COMMONBASE)/crypto/rsa/%.c
78 $(COMPILE.c) -o $@ $<
79 $(CTFCONVERT_O)
80 #
81 $(OBJS_DIR)/%.o: $(COMMONBASE)/bignum/%.c
82 $(COMPILE.c) -o $@ $<
83 $(CTFCONVERT_O)
84 #
85 $(OBJS_DIR)/%.o: $(UTSBASE)/common/bignum/%.c
86 $(COMPILE.c) -o $@ $<
87 $(CTFCONVERT_O)
88 #
89 $(OBJS_DIR)/%.o: $(COMMONBASE)/mpi/%.c
90 $(COMPILE.c) -o $@ $<
91 $(CTFCONVERT_O)
92 #
93 $(OBJS_DIR)/%.o: $(COMMONBASE)/acl/%.c
94 $(COMPILE.c) -o $@ $<
95 $(CTFCONVERT_O)
96 #
97 $(OBJS_DIR)/%.o: $(COMMONBASE)/avl/%.c
98 $(COMPILE.c) -o $@ $<
99 $(CTFCONVERT_O)
100 #
101 $(OBJS_DIR)/%.o: $(COMMONBASE)/ucode/%.c
102 $(COMPILE.c) -o $@ $<
103 $(CTFCONVERT_O)
104 #
105 $(OBJS_DIR)/%.o: $(UTSBASE)/common/brand/snl/%.c
106 $(COMPILE.c) -o $@ $<
107 $(CTFCONVERT_O)
108 #
109 $(OBJS_DIR)/%.o: $(UTSBASE)/common/brand/solaris10/%.c
110 $(COMPILE.c) -o $@ $<
111 $(CTFCONVERT_O)
112 #
113 $(OBJS_DIR)/%.o: $(UTSBASE)/common/c2/%.c
114 $(COMPILE.c) -o $@ $<
115 $(CTFCONVERT_O)
116 #
117 $(OBJS_DIR)/%.o: $(UTSBASE)/common/conf/%.c
118 $(COMPILE.c) -o $@ $<

```

new/usr/src/uts/common/Makefile.rules

```

119      $(CTFCONVERT_O)

121 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/contract/%.c
122   $(COMPILE.c) -o $@ $<
123   $(CTFCONVERT_O)

125 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/cpr/%.c
126   $(COMPILE.c) -o $@ $<
127   $(CTFCONVERT_O)

129 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ctf/%.c
130   $(COMPILE.c) -o $@ $<
131   $(CTFCONVERT_O)

133 $(OBJS_DIR)/%.o:      $(COMMONBASE)/ctf/%.c
134   $(COMPILE.c) -o $@ $<
135   $(CTFCONVERT_O)

137 $(OBJS_DIR)/%.o:      $(COMMONBASE)/crypto/des/%.c
138   $(COMPILE.c) -o $@ $<
139   $(CTFCONVERT_O)

141 $(OBJS_DIR)/%.o:      $(COMMONBASE)/secflags/%.c
142   $(COMPILE.c) -o $@ $<
143   $(CTFCONVERT_O)

145 $(OBJS_DIR)/%.o:      $(COMMONBASE)/smbios/%.c
146   $(COMPILE.c) -o $@ $<
147   $(CTFCONVERT_O)

149 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/des/%.c
150   $(COMPILE.c) -o $@ $<
151   $(CTFCONVERT_O)

153 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/crypto/api/%.c
154   $(COMPILE.c) -o $@ $<
155   $(CTFCONVERT_O)

157 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/crypto/core/%.c
158   $(COMPILE.c) -o $@ $<
159   $(CTFCONVERT_O)

161 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/crypto/io/%.c
162   $(COMPILE.c) -o $@ $<
163   $(CTFCONVERT_O)

165 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/crypto/spi/%.c
166   $(COMPILE.c) -o $@ $<
167   $(CTFCONVERT_O)

169 $(OBJS_DIR)/%.o:      $(COMMONBASE)/pci/%.c
170   $(COMPILE.c) -o $@ $<
171   $(CTFCONVERT_O)

173 $(OBJS_DIR)/%.o:      $(COMMONBASE)/devid/%.c
174   $(COMPILE.c) -o $@ $<
175   $(CTFCONVERT_O)

177 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/disp/%.c
178   $(COMPILE.c) -o $@ $<
179   $(CTFCONVERT_O)

181 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/dtrace/%.c
182   $(COMPILE.c) -o $@ $<
183   $(CTFCONVERT_O)

```

3

new/usr/src/uts/common/Makefile.rules

```

185 $(OBJS_DIR)/%.o:      $(COMMONBASE)/execct/%.c
186   $(COMPILE.c) -o $@ $<
187   $(CTFCONVERT_O)

189 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/exec/aout/%.c
190   $(COMPILE.c) -o $@ $<
191   $(CTFCONVERT_O)

193 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/exec/elf/%.c
194   $(COMPILE.c) -o $@ $<
195   $(CTFCONVERT_O)

197 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/exec/intp/%.c
198   $(COMPILE.c) -o $@ $<
199   $(CTFCONVERT_O)

201 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/exec/shbin/%.c
202   $(COMPILE.c) -o $@ $<
203   $(CTFCONVERT_O)

205 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/exec/java/%.c
206   $(COMPILE.c) -o $@ $<
207   $(CTFCONVERT_O)

209 $(OBJS_DIR)/$(FONT).c: $(FONT_DIR)/$(FONT_SRC).bdf
210   $(VTFONTCVT) -f source -o $@ $(FONT_DIR)/$(FONT_SRC).bdf

212 $(OBJS_DIR)/$(FONT).o: $(OBJS_DIR)/$(FONT).c
213   $(COMPILE.c) -o $@ $<
214   $(CTFCONVERT_O)

216 $(OBJS_DIR)/%.o:      $(SRC)/common/font/%.c
217   $(COMPILE.c) -o $@ $<
218   $(CTFCONVERT_O)

220 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/%.c
221   $(COMPILE.c) -o $@ $<
222   $(CTFCONVERT_O)

224 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/autofs/%.c
225   $(COMPILE.c) -o $@ $<
226   $(CTFCONVERT_O)

228 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/bootfs/%.c
229   $(COMPILE.c) -o $@ $<
230   $(CTFCONVERT_O)

232 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/dcfis/%.c
233   $(COMPILE.c) -o $@ $<
234   $(CTFCONVERT_O)

236 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/devfs/%.c
237   $(COMPILE.c) -o $@ $<
238   $(CTFCONVERT_O)

240 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/ctfs/%.c
241   $(COMPILE.c) -o $@ $<
242   $(CTFCONVERT_O)

244 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/doorfs/%.c
245   $(COMPILE.c) -o $@ $<
246   $(CTFCONVERT_O)

248 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/fs/dev/%.c
249   $(COMPILE.c) -o $@ $<
250   $(CTFCONVERT_O)

```

4

```

252 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/fd/%.c
253 $(COMPILE.c) -o $@ $<
254 $(CTFCONVERT_O)

256 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/fifofs/%.c
257 $(COMPILE.c) -o $@ $<
258 $(CTFCONVERT_O)

260 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/hsfs/%.c
261 $(COMPILE.c) -o $@ $<
262 $(CTFCONVERT_O)

264 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/lofs/%.c
265 $(COMPILE.c) -o $@ $<
266 $(CTFCONVERT_O)

268 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/mntfs/%.c
269 $(COMPILE.c) -o $@ $<
270 $(CTFCONVERT_O)

272 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/namefs/%.c
273 $(COMPILE.c) -o $@ $<
274 $(CTFCONVERT_O)

276 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/nfs/%.c
277 $(COMPILE.c) -o $@ $<
278 $(CTFCONVERT_O)

280 $(OBJSDIR)/%.o: $(COMMONBASE)/smbsrv/%.c
281 $(COMPILE.c) -o $@ $<
282 $(CTFCONVERT_O)

284 $(OBJSDIR)/%.o: $(COMMONBASE)/vga/%.c
285 $(COMPILE.c) -o $@ $<
286 $(CTFCONVERT_O)

288 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/smbsrv/%.c
289 $(COMPILE.c) -o $@ $<
290 $(CTFCONVERT_O)

292 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/objfs/%.c
293 $(COMPILE.c) -o $@ $<
294 $(CTFCONVERT_O)

296 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/pcfs/%.c
297 $(COMPILE.c) -o $@ $<
298 $(CTFCONVERT_O)

300 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/portfs/%.c
301 $(COMPILE.c) -o $@ $<
302 $(CTFCONVERT_O)

304 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/proc/%.c
305 $(COMPILE.c) -o $@ $<
306 $(CTFCONVERT_O)

308 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/sharefs/%.c
309 $(COMPILE.c) -o $@ $<
310 $(CTFCONVERT_O)

312 $(OBJSDIR)/%.o: $(COMMONBASE)/smbclnt/%.c
313 $(COMPILE.c) -o $@ $<
314 $(CTFCONVERT_O)

316 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/smbclnt/net smb/%.c

```

```

317 $(COMPILE.c) -o $@ $<
318 $(CTFCONVERT_O)

320 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/smbclnt/smbfs/%.c
321 $(COMPILE.c) -o $@ $<
322 $(CTFCONVERT_O)

324 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/sockfs/%.c
325 $(COMPILE.c) -o $@ $<
326 $(CTFCONVERT_O)

328 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/specfs/%.c
329 $(COMPILE.c) -o $@ $<
330 $(CTFCONVERT_O)

332 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/swapfs/%.c
333 $(COMPILE.c) -o $@ $<
334 $(CTFCONVERT_O)

336 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/tmpfs/%.c
337 $(COMPILE.c) -o $@ $<
338 $(CTFCONVERT_O)

340 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/udfs/%.c
341 $(COMPILE.c) -o $@ $<
342 $(CTFCONVERT_O)

344 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/ufs/%.c
345 $(COMPILE.c) -o $@ $<
346 $(CTFCONVERT_O)

348 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/vscan/%.c
349 $(COMPILE.c) -o $@ $<
350 $(CTFCONVERT_O)

352 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/zfs/%.c
353 $(COMPILE.c) -o $@ $<
354 $(CTFCONVERT_O)

356 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/zfs/luar/%.c
357 $(COMPILE.c) -o $@ $<
358 $(CTFCONVERT_O)

360 $(OBJSDIR)/%.o: $(UTSBASE)/common/fs/zut/%.c
361 $(COMPILE.c) -o $@ $<
362 $(CTFCONVERT_O)

364 $(OBJSDIR)/%.o: $(COMMONBASE)/xattr/%.c
365 $(COMPILE.c) -o $@ $<
366 $(CTFCONVERT_O)

368 $(OBJSDIR)/%.o: $(COMMONBASE)/zfs/%.c
369 $(COMPILE.c) -o $@ $<
370 $(CTFCONVERT_O)

372 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/adapters/pmcs/%.c
373 $(COMPILE.c) -o $@ $<
374 $(CTFCONVERT_O)

376 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/adapters/pmcs/%.bin
377 $(COMPILE.b) -o $@ $<
378 $(CTFCONVERT_O)

379 $(OBJSDIR)/%.o: $(COMMONBASE)/fsreparse/%.c
380 $(COMPILE.c) -o $@ $<
381 $(CTFCONVERT_O)

```

```

383 KMECHKRB5_BASE=$(UTSBASE)/common/gssapi/mechs/krb5
385 KGSSDFLAGS=-I $(UTSBASE)/common/gssapi/include

387 # Note, KRB5_DEFS can be assigned various preprocessor flags,
388 # typically -D defines on the make invocation. The standard compiler
389 # flags will not be overwritten.
390 KGSSDFLAGS += $(KRB5_DEFS)

392 $(OBJSDIR)/%.o:                $(UTSBASE)/common/gssapi/%.c
393     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
394     $(CTFCONVERT_O)

396 $(OBJSDIR)/%.o:                $(UTSBASE)/common/gssapi/mechs/dummy/%.c
397     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
398     $(CTFCONVERT_O)

400 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/%.c
401     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
402     $(CTFCONVERT_O)

404 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/%.c
405     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
406     $(CTFCONVERT_O)

408 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/des/%.c
409     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
410     $(CTFCONVERT_O)

412 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/arcfour/%.c
413     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
414     $(CTFCONVERT_O)

416 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/dk/%.c
417     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
418     $(CTFCONVERT_O)

420 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/enc_provider/%.c
421     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
422     $(CTFCONVERT_O)

424 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/hash_provider/%.c
425     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
426     $(CTFCONVERT_O)

428 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/keyhash_provider/%.c
429     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
430     $(CTFCONVERT_O)

432 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/raw/%.c
433     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
434     $(CTFCONVERT_O)

436 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/crypto/old/%.c
437     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
438     $(CTFCONVERT_O)

440 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/krb5/krb/%.c
441     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
442     $(CTFCONVERT_O)

444 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/krb5/os/%.c
445     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
446     $(CTFCONVERT_O)

```

```

448 $(OBJSDIR)/ser_sctx.o := CPPFLAGS += -DPROVIDE_KERNEL_IMPORT=1

450 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/mech/%.c
451     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
452     $(CTFCONVERT_O)

454 $(OBJSDIR)/%.o:                $(KMECHKRB5_BASE)/profile/%.c
455     $(COMPILE.c) $(KGSSDFLAGS) -o $$@ $<
456     $(CTFCONVERT_O)

458 $(OBJSDIR)/%.o:                $(UTSBASE)/common/idmap/%.c
459     $(COMPILE.c) -o $$@ $<
460     $(CTFCONVERT_O)

462 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/%.c
463     $(COMPILE.c) -o $$@ $<
464     $(CTFCONVERT_O)

466 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/arp/%.c
467     $(COMPILE.c) -o $$@ $<
468     $(CTFCONVERT_O)

470 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/ip/%.c
471     $(COMPILE.c) -o $$@ $<
472     $(CTFCONVERT_O)

474 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/ipnet/%.c
475     $(COMPILE.c) -o $$@ $<
476     $(CTFCONVERT_O)

478 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/iptun/%.c
479     $(COMPILE.c) -o $$@ $<
480     $(CTFCONVERT_O)

482 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/kssl/%.c
483     $(COMPILE.c) -o $$@ $<
484     $(CTFCONVERT_O)

486 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/sctp/%.c
487     $(COMPILE.c) -o $$@ $<
488     $(CTFCONVERT_O)

490 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/tcp/%.c
491     $(COMPILE.c) -o $$@ $<
492     $(CTFCONVERT_O)

494 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/ilb/%.c
495     $(COMPILE.c) -o $$@ $<
496     $(CTFCONVERT_O)

498 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/ipf/%.c
499     $(COMPILE.c) -o $$@ $<
500     $(CTFCONVERT_O)

502 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/ipd/%.c
503     $(COMPILE.c) -o $$@ $<
504     $(CTFCONVERT_O)

506 $(OBJSDIR)/%.o:                $(COMMONBASE)/net/patricia/%.c
507     $(COMPILE.c) -o $$@ $<
508     $(CTFCONVERT_O)

510 $(OBJSDIR)/%.o:                $(UTSBASE)/common/inet/udp/%.c
511     $(COMPILE.c) -o $$@ $<
512     $(CTFCONVERT_O)

```



```

514 $(OBJSDIR)/%.o: $(UTSBASE)/common/inet/nca/%.c
515 $(COMPILE.c) -o $@ $<
516 $(CTFCONVERT_O)

518 $(OBJSDIR)/%.o: $(UTSBASE)/common/inet/sockmods/%.c
519 $(COMPILE.c) -o $@ $<
520 $(CTFCONVERT_O)

522 $(OBJSDIR)/%.o: $(UTSBASE)/common/inet/dlpistub/%.c
523 $(COMPILE.c) -o $@ $<
524 $(CTFCONVERT_O)

526 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/%.c
527 $(COMPILE.c) -o $@ $<
528 $(CTFCONVERT_O)

530 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/l394/%.c
531 $(COMPILE.c) -o $@ $<
532 $(CTFCONVERT_O)

534 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/l394/adapters/%.c
535 $(COMPILE.c) -o $@ $<
536 $(CTFCONVERT_O)

538 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/l394/targets/avl394/%.c
539 $(COMPILE.c) -o $@ $<
540 $(CTFCONVERT_O)

542 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/l394/targets/dcaml394/%.c
543 $(COMPILE.c) -o $@ $<
544 $(CTFCONVERT_O)

546 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/l394/targets/scsal394/%.c
547 $(COMPILE.c) -o $@ $<
548 $(CTFCONVERT_O)

550 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sbp2/%.c
551 $(COMPILE.c) -o $@ $<
552 $(CTFCONVERT_O)

554 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/aac/%.c
555 $(COMPILE.c) -o $@ $<
556 $(CTFCONVERT_O)

558 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/afe/%.c
559 $(COMPILE.c) -o $@ $<
560 $(CTFCONVERT_O)

562 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/atge/%.c
563 $(COMPILE.c) -o $@ $<
564 $(CTFCONVERT_O)

566 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/arn/%.c
567 $(COMPILE.c) -o $@ $<
568 $(CTFCONVERT_O)

570 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ath/%.c
571 $(COMPILE.c) -o $@ $<
572 $(CTFCONVERT_O)

574 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/atu/%.c
575 $(COMPILE.c) -o $@ $<
576 $(CTFCONVERT_O)

578 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/impl/%.c
579 $(COMPILE.c) -o $@ $<

```

```

580 $(CTFCONVERT_O)

582 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/ac97/%.c
583 $(COMPILE.c) -o $@ $<
584 $(CTFCONVERT_O)

586 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audioens/%.c
587 $(COMPILE.c) -o $@ $<
588 $(CTFCONVERT_O)

590 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audioemu10k/%.c
591 $(COMPILE.c) -o $@ $<
592 $(CTFCONVERT_O)

594 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audio1575/%.c
595 $(COMPILE.c) -o $@ $<
596 $(CTFCONVERT_O)

598 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audio810/%.c
599 $(COMPILE.c) -o $@ $<
600 $(CTFCONVERT_O)

602 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiocmi/%.c
603 $(COMPILE.c) -o $@ $<
604 $(CTFCONVERT_O)

606 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiocmhd/%.c
607 $(COMPILE.c) -o $@ $<
608 $(CTFCONVERT_O)

610 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiohd/%.c
611 $(COMPILE.c) -o $@ $<
612 $(CTFCONVERT_O)

614 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audioixp/%.c
615 $(COMPILE.c) -o $@ $<
616 $(CTFCONVERT_O)

618 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiols/%.c
619 $(COMPILE.c) -o $@ $<
620 $(CTFCONVERT_O)

622 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiopci/%.c
623 $(COMPILE.c) -o $@ $<
624 $(CTFCONVERT_O)

626 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiopl6x/%.c
627 $(COMPILE.c) -o $@ $<
628 $(CTFCONVERT_O)

630 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiosolo/%.c
631 $(COMPILE.c) -o $@ $<
632 $(CTFCONVERT_O)

634 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiotots/%.c
635 $(COMPILE.c) -o $@ $<
636 $(CTFCONVERT_O)

638 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/audio/drv/audiovia823x/%.c
639 $(COMPILE.c) -o $@ $<
640 $(CTFCONVERT_O)

642 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bfe/%.c
643 $(COMPILE.c) -o $@ $<
644 $(CTFCONVERT_O)

```

```

646 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bge/%.c
647 $(COMPILE.c) -o $@ $<
648 $(CTFCONVERT_O)

650 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/blkdev/%.c
651 $(COMPILE.c) -o $@ $<
652 $(CTFCONVERT_O)

654 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnx/%.c
655 $(COMPILE.c) -o $@ $<
656 $(CTFCONVERT_O)

658 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnx/570x/driver/common/lmde
659 $(COMPILE.c) -o $@ $<
660 $(CTFCONVERT_O)

662 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/%.c
663 $(COMPILE.c) -o $@ $<
664 $(CTFCONVERT_O)

666 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/577xx/common/%.c
667 $(COMPILE.c) -o $@ $<
668 $(CTFCONVERT_O)

670 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/e
671 $(COMPILE.c) -o $@ $<
672 $(CTFCONVERT_O)

674 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
675 $(COMPILE.c) -o $@ $<
676 $(CTFCONVERT_O)

678 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
679 $(COMPILE.c) -o $@ $<
680 $(CTFCONVERT_O)

682 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
683 $(COMPILE.c) -o $@ $<
684 $(CTFCONVERT_O)

686 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
687 $(COMPILE.c) -o $@ $<
688 $(CTFCONVERT_O)

690 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
691 $(COMPILE.c) -o $@ $<
692 $(CTFCONVERT_O)

694 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/bpf/%.c
695 $(COMPILE.c) -o $@ $<
696 $(CTFCONVERT_O)

698 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/cardbus/%.c
699 $(COMPILE.c) -o $@ $<
700 $(CTFCONVERT_O)

702 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/stmf/%.c
703 $(COMPILE.c) -o $@ $<
704 $(CTFCONVERT_O)

706 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/port/fct/%.c
707 $(COMPILE.c) -o $@ $<
708 $(CTFCONVERT_O)

710 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/port/qlt/%.c
711 $(COMPILE.c) -o $@ $<

```

```

712 $(CTFCONVERT_O)

714 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/port/srpt/%.c
715 $(COMPILE.c) -o $@ $<
716 $(CTFCONVERT_O)

718 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/port/fcoet/%.c
719 $(COMPILE.c) -o $@ $<
720 $(CTFCONVERT_O)

722 $(OBJSDIR)/%.o: $(COMMONBASE)/iscsit/%.c
723 $(COMPILE.c) -o $@ $<
724 $(CTFCONVERT_O)

726 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/port/iscsit/%.c
727 $(COMPILE.c) -o $@ $<
728 $(CTFCONVERT_O)

730 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/port/pppt/%.c
731 $(COMPILE.c) -o $@ $<
732 $(CTFCONVERT_O)

734 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/comstar/lu/stmf_sbd/%.c
735 $(COMPILE.c) -o $@ $<
736 $(CTFCONVERT_O)

738 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/cpqary3/%.c
739 $(COMPILE.c) -o $@ $<
740 $(CTFCONVERT_O)

742 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/dld/%.c
743 $(COMPILE.c) -o $@ $<
744 $(CTFCONVERT_O)

746 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/dls/%.c
747 $(COMPILE.c) -o $@ $<
748 $(CTFCONVERT_O)

750 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/dmfe/%.c
751 $(COMPILE.c) -o $@ $<
752 $(CTFCONVERT_O)

754 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/drm/%.c
755 $(COMPILE.c) -o $@ $<
756 $(CTFCONVERT_O)

758 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/efe/%.c
759 $(COMPILE.c) -o $@ $<
760 $(CTFCONVERT_O)

762 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/elxl/%.c
763 $(COMPILE.c) -o $@ $<
764 $(CTFCONVERT_O)

766 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fcoe/%.c
767 $(COMPILE.c) -o $@ $<
768 $(CTFCONVERT_O)

770 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/hme/%.c
771 $(COMPILE.c) -o $@ $<
772 $(CTFCONVERT_O)

774 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/pciex/%.c
775 $(COMPILE.c) -o $@ $<
776 $(CTFCONVERT_O)

```

```

778 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/hotplug/hpcsvc/%.c
779 $(COMPILE.c) -o $@ $<
780 $(CTFCONVERT_O)

782 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/pciex/hotplug/%.c
783 $(COMPILE.c) -o $@ $<
784 $(CTFCONVERT_O)

786 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/hotplug/pcihp/%.c
787 $(COMPILE.c) -o $@ $<
788 $(CTFCONVERT_O)

790 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/rds/%.c
791 $(COMPILE.c) -o $@ $<
792 $(CTFCONVERT_O)

794 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/rdsv3/%.c
795 $(COMPILE.c) -o $@ $<
796 $(CTFCONVERT_O)

798 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/iser/%.c
799 $(COMPILE.c) -o $@ $<
800 $(CTFCONVERT_O)

802 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/ibd/%.c
803 $(COMPILE.c) -o $@ $<
804 $(CTFCONVERT_O)

806 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/eoib/%.c
807 $(COMPILE.c) -o $@ $<
808 $(CTFCONVERT_O)

810 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/of/sol_ofs/%.c
811 $(COMPILE.c) -o $@ $<
812 $(CTFCONVERT_O)

814 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/of/sol_ucma/%.c
815 $(COMPILE.c) -o $@ $<
816 $(CTFCONVERT_O)

818 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/of/sol_umad/%.c
819 $(COMPILE.c) -o $@ $<
820 $(CTFCONVERT_O)

822 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/of/sol_uverbs/%.
823 $(COMPILE.c) -o $@ $<
824 $(CTFCONVERT_O)

826 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/sdp/%.c
827 $(COMPILE.c) -o $@ $<
828 $(CTFCONVERT_O)

830 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/mgt/ibcm/%.c
831 $(COMPILE.c) -o $@ $<
832 $(CTFCONVERT_O)

834 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/mgt/ibdm/%.c
835 $(COMPILE.c) -o $@ $<
836 $(CTFCONVERT_O)

838 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/mgt/ibdma/%.c
839 $(COMPILE.c) -o $@ $<
840 $(CTFCONVERT_O)

842 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/mgt/ibmf/%.c
843 $(COMPILE.c) -o $@ $<

```

```

844 $(CTFCONVERT_O)

846 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/ibnex/%.c
847 $(COMPILE.c) -o $@ $<
848 $(CTFCONVERT_O)

850 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/ibt1/%.c
851 $(COMPILE.c) -o $@ $<
852 $(CTFCONVERT_O)

854 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/adapters/tavor/%.c
855 $(COMPILE.c) -o $@ $<
856 $(CTFCONVERT_O)

858 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/adapters/hermon/%.c
859 $(COMPILE.c) -o $@ $<
860 $(CTFCONVERT_O)

862 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ib/clients/daplt/%.c
863 $(COMPILE.c) -o $@ $<
864 $(CTFCONVERT_O)

866 $(OBJSDIR)/%.o: $(COMMONBASE)/iscsi/%.c
867 $(COMPILE.c) -o $@ $<
868 $(CTFCONVERT_O)

870 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/idm/%.c
871 $(COMPILE.c) -o $@ $<
872 $(CTFCONVERT_O)

874 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ipw/%.c
875 $(COMPILE.c) -o $@ $<
876 $(CTFCONVERT_O)

878 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/iwh/%.c
879 $(COMPILE.c) -o $@ $<
880 $(CTFCONVERT_O)

882 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/iwi/%.c
883 $(COMPILE.c) -o $@ $<
884 $(CTFCONVERT_O)

886 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/iwk/%.c
887 $(COMPILE.c) -o $@ $<
888 $(CTFCONVERT_O)

890 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/iwn/%.c
891 $(COMPILE.c) -o $@ $<
892 $(CTFCONVERT_O)

894 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/iwp/%.c
895 $(COMPILE.c) -o $@ $<
896 $(CTFCONVERT_O)

898 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/kb8042/%.c
899 $(COMPILE.c) -o $@ $<
900 $(CTFCONVERT_O)

902 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/kbtrans/%.c
903 $(COMPILE.c) -o $@ $<
904 $(CTFCONVERT_O)

906 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ksocket/%.c
907 $(COMPILE.c) -o $@ $<
908 $(CTFCONVERT_O)

```

```

910 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/aggr/%.c
911     $(COMPILE.c) -o $@ $<
912     $(CTFCONVERT_O)

914 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/lp/%.c
915     $(COMPILE.c) -o $@ $<
916     $(CTFCONVERT_O)

918 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mac/%.c
919     $(COMPILE.c) -o $@ $<
920     $(CTFCONVERT_O)

922 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mac/plugins/%.c
923     $(COMPILE.c) -o $@ $<
924     $(CTFCONVERT_O)

926 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mega_sas/%.c
927     $(COMPILE.c) -o $@ $<
928     $(CTFCONVERT_O)

930 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mii/%.c
931     $(COMPILE.c) -o $@ $<
932     $(CTFCONVERT_O)

934 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mr_sas/%.c
935     $(COMPILE.c) -o $@ $<
936     $(CTFCONVERT_O)

938 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/adapters/mpt_sas/%.c
939     $(COMPILE.c) -o $@ $<
940     $(CTFCONVERT_O)

942 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mxfe/%.c
943     $(COMPILE.c) -o $@ $<
944     $(CTFCONVERT_O)

946 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mwl/%.c
947     $(COMPILE.c) -o $@ $<
948     $(CTFCONVERT_O)

950 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/mwl/mwl_fw/%.c
951     $(COMPILE.c) -o $@ $<
952     $(CTFCONVERT_O)

954 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/net80211/%.c
955     $(COMPILE.c) -o $@ $<
956     $(CTFCONVERT_O)

958 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/nge/%.c
959     $(COMPILE.c) -o $@ $<
960     $(CTFCONVERT_O)

962 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/nvme/%.c
963     $(COMPILE.c) -o $@ $<
964     $(CTFCONVERT_O)

966 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/nxge/%.c
967     $(COMPILE.c) -o $@ $<
968     $(CTFCONVERT_O)

970 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/nxge/np/%.c
971     $(COMPILE.c) -o $@ $<
972     $(CTFCONVERT_O)

974 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/nxge/%.s
975     $(COMPILE.s) -o $@ $<

```

```

977 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/pci-ide/%.c
978     $(COMPILE.c) -o $@ $<
979     $(CTFCONVERT_O)

981 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/pcn/%.c
982     $(COMPILE.c) -o $@ $<
983     $(CTFCONVERT_O)

985 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ppp/sppp/%.c
986     $(COMPILE.c) -o $@ $<
987     $(CTFCONVERT_O)

989 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ppp/spppasyn/%.c
990     $(COMPILE.c) -o $@ $<
991     $(CTFCONVERT_O)

993 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ppp/sppptun/%.c
994     $(COMPILE.c) -o $@ $<
995     $(CTFCONVERT_O)

997 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/qede/%.c
998     $(COMPILE.c) -o $@ $<
999     $(CTFCONVERT_O)

1001 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/qede/579xx/drivers/ecore/%.
1002     $(COMPILE.c) -o $@ $<
1003     $(CTFCONVERT_O)

1005 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ral/%.c
1006     $(COMPILE.c) -o $@ $<
1007     $(CTFCONVERT_O)

1009 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/rge/%.c
1010     $(COMPILE.c) -o $@ $<
1011     $(CTFCONVERT_O)

1013 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/rtls/%.c
1014     $(COMPILE.c) -o $@ $<
1015     $(CTFCONVERT_O)

1017 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/rsm/%.c
1018     $(COMPILE.c) -o $@ $<
1019     $(CTFCONVERT_O)

1021 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/rtw/%.c
1022     $(COMPILE.c) -o $@ $<
1023     $(CTFCONVERT_O)

1025 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/rum/%.c
1026     $(COMPILE.c) -o $@ $<
1027     $(CTFCONVERT_O)

1029 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/rwd/%.c
1030     $(COMPILE.c) -o $@ $<
1031     $(CTFCONVERT_O)

1033 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/rwn/%.c
1034     $(COMPILE.c) -o $@ $<
1035     $(CTFCONVERT_O)

1037 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sata/adapters/ahci/%.c
1038     $(COMPILE.c) -o $@ $<
1039     $(CTFCONVERT_O)

1041 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sata/adapters/nv_sata/%.c

```

```

1042 $(COMPILE.c) -o $$ $<
1043 $(CTFCONVERT_O)

1045 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sata/adapters/si3124/%.c
1046 $(COMPILE.c) -o $$ $<
1047 $(CTFCONVERT_O)

1049 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sata/impl/%.c
1050 $(COMPILE.c) -o $$ $<
1051 $(CTFCONVERT_O)

1053 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/conf/%.c
1054 $(COMPILE.c) -o $$ $<
1055 $(CTFCONVERT_O)

1057 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/impl/%.c
1058 $(COMPILE.c) -o $$ $<
1059 $(CTFCONVERT_O)

1061 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/targets/%.c
1062 $(COMPILE.c) -o $$ $<
1063 $(CTFCONVERT_O)

1065 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/adapters/%.c
1066 $(COMPILE.c) -o $$ $<
1067 $(CTFCONVERT_O)

1069 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/adapters/blk2scsa/%.c
1070 $(COMPILE.c) -o $$ $<
1071 $(CTFCONVERT_O)

1073 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/adapters/scsi_vhci/%.c
1074 $(COMPILE.c) -o $$ $<
1075 $(CTFCONVERT_O)

1077 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/scsi/adapters/scsi_vhci/fop
1078 $(COMPILE.c) -o $$ $<
1079 $(CTFCONVERT_O)

1081 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fibre-channel/ulp/%.c
1082 $(COMPILE.c) -o $$ $<
1083 $(CTFCONVERT_O)

1085 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fibre-channel/impl/%.c
1086 $(COMPILE.c) -o $$ $<
1087 $(CTFCONVERT_O)

1089 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fibre-channel/fca/qlc/%.c
1090 $(COMPILE.c) -o $$ $<
1091 $(CTFCONVERT_O)

1093 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fibre-channel/fca/qlge/%.c
1094 $(COMPILE.c) -o $$ $<
1095 $(CTFCONVERT_O)

1097 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fibre-channel/fca/emlxs/%.c
1098 $(COMPILE.c) -o $$ $<
1099 $(CTFCONVERT_O)

1101 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fibre-channel/fca/oce/%.c
1102 $(COMPILE.c) -o $$ $<
1103 $(CTFCONVERT_O)

1105 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/fibre-channel/fca/fcoei/%.c
1106 $(COMPILE.c) -o $$ $<
1107 $(CTFCONVERT_O)

```

```

1109 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sdcard/adapters/sdhost/%.c
1110 $(COMPILE.c) -o $$ $<
1111 $(CTFCONVERT_O)

1113 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sdcard/impl/%.c
1114 $(COMPILE.c) -o $$ $<
1115 $(CTFCONVERT_O)

1117 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sdcard/targets/sdcard/%.c
1118 $(COMPILE.c) -o $$ $<
1119 $(CTFCONVERT_O)

1121 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sfe/%.c
1122 $(COMPILE.c) -o $$ $<
1123 $(CTFCONVERT_O)

1125 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/simnet/%.c
1126 $(COMPILE.c) -o $$ $<
1127 $(CTFCONVERT_O)

1129 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/softmac/%.c
1130 $(COMPILE.c) -o $$ $<
1131 $(CTFCONVERT_O)

1133 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/uath/%.c
1134 $(COMPILE.c) -o $$ $<
1135 $(CTFCONVERT_O)

1137 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/uath/uath_fw/%.c
1138 $(COMPILE.c) -o $$ $<
1139 $(CTFCONVERT_O)

1141 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/ural/%.c
1142 $(COMPILE.c) -o $$ $<
1143 $(CTFCONVERT_O)

1145 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/urtw/%.c
1146 $(COMPILE.c) -o $$ $<
1147 $(CTFCONVERT_O)

1149 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/audio/usb_ac/%.
1150 $(COMPILE.c) -o $$ $<
1151 $(CTFCONVERT_O)

1153 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/audio/usb_as/%.
1154 $(COMPILE.c) -o $$ $<
1155 $(CTFCONVERT_O)

1157 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/audio/usb_ah/%.
1158 $(COMPILE.c) -o $$ $<
1159 $(CTFCONVERT_O)

1161 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbskel/%.c
1162 $(COMPILE.c) -o $$ $<
1163 $(CTFCONVERT_O)

1165 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/video/usbvc/%.c
1166 $(COMPILE.c) -o $$ $<
1167 $(CTFCONVERT_O)

1169 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/hid/%.c
1170 $(COMPILE.c) -o $$ $<
1171 $(CTFCONVERT_O)

1173 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/hidparser/%.c

```

```

1174 $(COMPILE.c) -o $$@ $<
1175 $(CTFCONVERT_O)

1177 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/printer/%.c
1178 $(COMPILE.c) -o $$@ $<
1179 $(CTFCONVERT_O)

1181 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbkbm/%.c
1182 $(COMPILE.c) -o $$@ $<
1183 $(CTFCONVERT_O)

1185 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbms/%.c
1186 $(COMPILE.c) -o $$@ $<
1187 $(CTFCONVERT_O)

1189 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbinput/usbwcm
1190 $(COMPILE.c) -o $$@ $<
1191 $(CTFCONVERT_O)

1193 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/ugen/%.c
1194 $(COMPILE.c) -o $$@ $<
1195 $(CTFCONVERT_O)

1197 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbser/%.c
1198 $(COMPILE.c) -o $$@ $<
1199 $(CTFCONVERT_O)

1201 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbser/usbsacm/
1202 $(COMPILE.c) -o $$@ $<
1203 $(CTFCONVERT_O)

1205 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbser/usbftdi/
1206 $(COMPILE.c) -o $$@ $<
1207 $(CTFCONVERT_O)

1209 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbser/usbser_k
1210 $(COMPILE.c) -o $$@ $<
1211 $(CTFCONVERT_O)

1213 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbser/usbsprl/
1214 $(COMPILE.c) -o $$@ $<
1215 $(CTFCONVERT_O)

1217 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/clients/usbecm/%.c
1218 $(COMPILE.c) -o $$@ $<
1219 $(CTFCONVERT_O)

1221 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/hcd/openhci/%.c
1222 $(COMPILE.c) -o $$@ $<
1223 $(CTFCONVERT_O)

1225 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/hcd/ehci/%.c
1226 $(COMPILE.c) -o $$@ $<
1227 $(CTFCONVERT_O)

1229 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/hcd/xhci/%.c
1230 $(COMPILE.c) -o $$@ $<
1231 $(CTFCONVERT_O)

1233 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/hcd/uhci/%.c
1234 $(COMPILE.c) -I../common -o $$@ $<
1235 $(CTFCONVERT_O)

1237 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/hubd/%.c
1238 $(COMPILE.c) -o $$@ $<
1239 $(CTFCONVERT_O)

```

```

1241 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/scsa2usb/%.c
1242 $(COMPILE.c) -o $$@ $<
1243 $(CTFCONVERT_O)

1245 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/usb_mid/%.c
1246 $(COMPILE.c) -o $$@ $<
1247 $(CTFCONVERT_O)

1249 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/usb_ia/%.c
1250 $(COMPILE.c) -o $$@ $<
1251 $(CTFCONVERT_O)

1253 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/usba/%.c
1254 $(COMPILE.c) -o $$@ $<
1255 $(CTFCONVERT_O)

1257 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usb/usba10/%.c
1258 $(COMPILE.c) -o $$@ $<
1259 $(CTFCONVERT_O)

1261 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/usbgem/%.c
1262 $(COMPILE.c) -o $$@ $<
1263 $(CTFCONVERT_O)

1265 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/axf/%.c
1266 $(COMPILE.c) -o $$@ $<
1267 $(CTFCONVERT_O)

1269 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/udf/%.c
1270 $(COMPILE.c) -o $$@ $<
1271 $(CTFCONVERT_O)

1273 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/udmf/%.c
1274 $(COMPILE.c) -o $$@ $<
1275 $(CTFCONVERT_O)

1277 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/upf/%.c
1278 $(COMPILE.c) -o $$@ $<
1279 $(CTFCONVERT_O)

1281 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/urf/%.c
1282 $(COMPILE.c) -o $$@ $<
1283 $(CTFCONVERT_O)

1285 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/vuidmice/%.c
1286 $(COMPILE.c) -o $$@ $<
1287 $(CTFCONVERT_O)

1289 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/vnic/%.c
1290 $(COMPILE.c) -o $$@ $<
1291 $(CTFCONVERT_O)

1293 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/wpi/%.c
1294 $(COMPILE.c) -o $$@ $<
1295 $(CTFCONVERT_O)

1297 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/zyd/%.c
1298 $(COMPILE.c) -o $$@ $<
1299 $(CTFCONVERT_O)

1301 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/chxge/com/%.c
1302 $(COMPILE.c) -o $$@ $<
1303 $(CTFCONVERT_O)

1305 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/chxge/%.c

```

```

1306      $(COMPILE.c) -o $$@ $<
1307      $(CTFCONVERT_O)

1309 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/cxgbe/common/%.c
1310      $(COMPILE.c) -o $$@ $<
1311      $(CTFCONVERT_O)

1313 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/cxgbe/shared/%.c
1314      $(COMPILE.c) -o $$@ $<
1315      $(CTFCONVERT_O)

1317 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/cxgbe/firmware/%.c
1318      $(COMPILE.c) -o $$@ $<
1319      $(CTFCONVERT_O)

1321 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/cxgbe/t4nex/%.c
1322      $(COMPILE.c) -o $$@ $<
1323      $(CTFCONVERT_O)

1325 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/cxgbe/cxgbe/%.c
1326      $(COMPILE.c) -o $$@ $<
1327      $(CTFCONVERT_O)

1329 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/ixgb/%.c
1330      $(COMPILE.c) -o $$@ $<
1331      $(CTFCONVERT_O)

1333 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/xge/drv/%.c
1334      $(COMPILE.c) -o $$@ $<
1335      $(CTFCONVERT_O)

1337 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/xge/hal/xgehal/%.c
1338      $(COMPILE.c) -o $$@ $<
1339      $(CTFCONVERT_O)

1341 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/e1000api/%.c
1342      $(COMPILE.c) -o $$@ $<
1343      $(CTFCONVERT_O)

1345 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/e1000g/%.c
1346      $(COMPILE.c) -o $$@ $<
1347      $(CTFCONVERT_O)

1349 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/igb/%.c
1350      $(COMPILE.c) -o $$@ $<
1351      $(CTFCONVERT_O)

1353 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/iprb/%.c
1354      $(COMPILE.c) -o $$@ $<
1355      $(CTFCONVERT_O)

1357 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/ixgbe/%.c
1358      $(COMPILE.c) -o $$@ $<
1359      $(CTFCONVERT_O)

1361 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/ixgbe/core/%.c
1362      $(COMPILE.c) -o $$@ $<
1363      $(CTFCONVERT_O)

1365 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/i40e/%.c
1366      $(COMPILE.c) -o $$@ $<
1367      $(CTFCONVERT_O)

1369 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/i40e/core/%.c
1370      $(COMPILE.c) -o $$@ $<
1371      $(CTFCONVERT_O)

```

```

1373 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/ntxn/%.c
1374      $(COMPILE.c) -o $$@ $<
1375      $(CTFCONVERT_O)

1377 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/myril0ge/drv/%.c
1378      $(COMPILE.c) -o $$@ $<
1379      $(CTFCONVERT_O)

1381 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ipp/%.c
1382      $(COMPILE.c) -o $$@ $<
1383      $(CTFCONVERT_O)

1385 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ipp/ipgpc/%.c
1386      $(COMPILE.c) -o $$@ $<
1387      $(CTFCONVERT_O)

1389 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ipp/dlcosmk/%.c
1390      $(COMPILE.c) -o $$@ $<
1391      $(CTFCONVERT_O)

1393 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ipp/flowacct/%.c
1394      $(COMPILE.c) -o $$@ $<
1395      $(CTFCONVERT_O)

1397 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ipp/dscpmk/%.c
1398      $(COMPILE.c) -o $$@ $<
1399      $(CTFCONVERT_O)

1401 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ipp/meters/%.c
1402      $(COMPILE.c) -o $$@ $<
1403      $(CTFCONVERT_O)

1405 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/kiconv/kiconv_emea/%.c
1406      $(COMPILE.c) -o $$@ $<
1407      $(CTFCONVERT_O)

1409 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/kiconv/kiconv_ja/%.c
1410      $(COMPILE.c) -o $$@ $<
1411      $(CTFCONVERT_O)

1413 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/kiconv/kiconv_ko/%.c
1414      $(COMPILE.c) -o $$@ $<
1415      $(CTFCONVERT_O)

1417 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/kiconv/kiconv_sc/%.c
1418      $(COMPILE.c) -o $$@ $<
1419      $(CTFCONVERT_O)

1421 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/kiconv/kiconv_tc/%.c
1422      $(COMPILE.c) -o $$@ $<
1423      $(CTFCONVERT_O)

1425 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/klm/%.c
1426      $(COMPILE.c) -o $$@ $<
1427      $(CTFCONVERT_O)

1429 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/kmdb/%.c
1430      $(COMPILE.c) -o $$@ $<
1431      $(CTFCONVERT_O)

1433 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/ktli/%.c
1434      $(COMPILE.c) -o $$@ $<
1435      $(CTFCONVERT_O)

1437 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/scsi/adapters/iscsi/%.c

```

```

1438 $(COMPILE.c) -o $@ $<
1439 $(CTFCONVERT_O)

1441 $(OBJSDIR)/%.o: $(COMMONBASE)/iscsi/%.c
1442 $(COMPILE.c) -o $@ $<
1443 $(CTFCONVERT_O)

1445 $(OBJSDIR)/%.o: $(UTSBASE)/common/inet/kifconf/%.c
1446 $(COMPILE.c) -o $@ $<
1447 $(CTFCONVERT_O)

1449 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/vr/%.c
1450 $(COMPILE.c) -o $@ $<
1451 $(CTFCONVERT_O)

1453 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/yge/%.c
1454 $(COMPILE.c) -o $@ $<
1455 $(CTFCONVERT_O)

1457 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sfxge/%.c
1458 $(COMPILE.c) -o $@ $<
1459 $(CTFCONVERT_O)

1461 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/sfxge/common/%.c
1462 $(COMPILE.c) -o $@ $<
1463 $(CTFCONVERT_O)

1465 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/skd/%.c
1466 $(COMPILE.c) -o $@ $<
1467 $(CTFCONVERT_O)

1469 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/virtio/%.c
1470 $(COMPILE.c) -o $@ $<
1471 $(CTFCONVERT_O)

1473 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/vioblk/%.c
1474 $(COMPILE.c) -o $@ $<
1475 $(CTFCONVERT_O)

1477 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/vioif/%.c
1478 $(COMPILE.c) -o $@ $<
1479 $(CTFCONVERT_O)
1480 #
1481 # krtld must refer to its own bzero/bcopy until the kernel is fully linked
1482 #
1483 $(OBJSDIR)/bootrd.o := CPPFLAGS += -DKOBJ_OVERRIDES
1484 $(OBJSDIR)/doreloc.o := CPPFLAGS += -DKOBJ_OVERRIDES
1485 $(OBJSDIR)/kobj.o := CPPFLAGS += -DKOBJ_OVERRIDES
1486 $(OBJSDIR)/kobj_boot.o := CPPFLAGS += -DKOBJ_OVERRIDES
1487 $(OBJSDIR)/kobj_bootflags.o := CPPFLAGS += -DKOBJ_OVERRIDES
1488 $(OBJSDIR)/kobj_convrelstr.o := CPPFLAGS += -DKOBJ_OVERRIDES
1489 $(OBJSDIR)/kobj_isa.o := CPPFLAGS += -DKOBJ_OVERRIDES
1490 $(OBJSDIR)/kobj_kdi.o := CPPFLAGS += -DKOBJ_OVERRIDES
1491 $(OBJSDIR)/kobj_lm.o := CPPFLAGS += -DKOBJ_OVERRIDES
1492 $(OBJSDIR)/kobj_reloc.o := CPPFLAGS += -DKOBJ_OVERRIDES
1493 $(OBJSDIR)/kobj_stubs.o := CPPFLAGS += -DKOBJ_OVERRIDES
1494 $(OBJSDIR)/kobj_subr.o := CPPFLAGS += -DKOBJ_OVERRIDES

1496 $(OBJSDIR)/%.o: $(UTSBASE)/common/krtld/%.c
1497 $(COMPILE.c) -o $@ $<
1498 $(CTFCONVERT_O)

1500 $(OBJSDIR)/%.o: $(COMMONBASE)/list/%.c
1501 $(COMPILE.c) -o $@ $<
1502 $(CTFCONVERT_O)

```

```

1504 $(OBJSDIR)/%.o: $(COMMONBASE)/lzma/%.c
1505 $(COMPILE.c) -o $@ $<
1506 $(CTFCONVERT_O)

1508 $(OBJSDIR)/%.o: $(COMMONBASE)/crypto/md4/%.c
1509 $(COMPILE.c) -o $@ $<
1510 $(CTFCONVERT_O)

1512 $(OBJSDIR)/%.o: $(COMMONBASE)/crypto/md5/%.c
1513 $(COMPILE.c) -o $@ $<
1514 $(CTFCONVERT_O)

1516 $(OBJSDIR)/%.o: $(COMMONBASE)/net/dhcp/%.c
1517 $(COMPILE.c) -o $@ $<
1518 $(CTFCONVERT_O)

1520 $(OBJSDIR)/%.o: $(COMMONBASE)/nvpair/%.c
1521 $(COMPILE.c) -o $@ $<
1522 $(CTFCONVERT_O)

1524 $(OBJSDIR)/%.o: $(UTSBASE)/common/os/%.c
1525 $(COMPILE.c) -o $@ $<
1526 $(CTFCONVERT_O)

1528 $(OBJSDIR)/%.o: $(UTSBASE)/common/pcmcia/cis/%.c
1529 $(COMPILE.c) -o $@ $<
1530 $(CTFCONVERT_O)

1532 $(OBJSDIR)/%.o: $(UTSBASE)/common/pcmcia/cs/%.c
1533 $(COMPILE.c) -o $@ $<
1534 $(CTFCONVERT_O)

1536 $(OBJSDIR)/%.o: $(UTSBASE)/common/pcmcia/nexus/%.c
1537 $(COMPILE.c) -o $@ $<
1538 $(CTFCONVERT_O)

1540 $(OBJSDIR)/%.o: $(UTSBASE)/common/pcmcia/pcs/%.c
1541 $(COMPILE.c) -o $@ $<
1542 $(CTFCONVERT_O)

1544 $(OBJSDIR)/%.o: $(UTSBASE)/common/rpc/%.c
1545 $(COMPILE.c) -o $@ $<
1546 $(CTFCONVERT_O)

1548 $(OBJSDIR)/%.o: $(UTSBASE)/common/rpc/sec/%.c
1549 $(COMPILE.c) -o $@ $<
1550 $(CTFCONVERT_O)

1552 $(OBJSDIR)/%.o: $(UTSBASE)/common/rpc/sec_gss/%.c
1553 $(COMPILE.c) -o $@ $<
1554 $(CTFCONVERT_O)

1556 $(OBJSDIR)/%.o: $(COMMONBASE)/crypto/edonr/%.c
1557 $(COMPILE.c) -o $@ $<
1558 $(CTFCONVERT_O)

1560 $(OBJSDIR)/%.o: $(COMMONBASE)/crypto/shal/%.c
1561 $(COMPILE.c) -o $@ $<
1562 $(CTFCONVERT_O)

1564 $(OBJSDIR)/%.o: $(COMMONBASE)/crypto/sha2/%.c
1565 $(COMPILE.c) -o $@ $<
1566 $(CTFCONVERT_O)

1568 $(OBJSDIR)/%.o: $(COMMONBASE)/crypto/skein/%.c
1569 $(COMPILE.c) -o $@ $<

```



```

1570      $(CTFCONVERT_O)

1572 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/syscall/%.c
1573     $(COMPILE.c) -o $@ $<
1574     $(CTFCONVERT_O)

1576 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/tnf/%.c
1577     $(COMPILE.c) -o $@ $<
1578     $(CTFCONVERT_O)

1580 $(OBJS_DIR)/%.o:      $(COMMONBASE)/tsol/%.c
1581     $(COMPILE.c) -o $@ $<
1582     $(CTFCONVERT_O)

1584 $(OBJS_DIR)/%.o:      $(COMMONBASE)/util/%.c
1585     $(COMPILE.c) -o $@ $<
1586     $(CTFCONVERT_O)

1588 $(OBJS_DIR)/%.o:      $(COMMONBASE)/unicode/%.c
1589     $(COMPILE.c) -o $@ $<
1590     $(CTFCONVERT_O)

1592 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/vm/%.c
1593     $(COMPILE.c) -o $@ $<
1594     $(CTFCONVERT_O)

1596 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/zmod/%.c
1597     $(COMPILE.c) -o $@ $<
1598     $(CTFCONVERT_O)

1600 $(OBJS_DIR)/zlib_obj.o:  $(ZLIB_OBJS:%=$(OBJS_DIR)/%)
1601     $(LD) -r -Breduce -M$(UTSBASE)/common/zmod/mapfile -o $@ \
1602     $(ZLIB_OBJS:%=$(OBJS_DIR)/%)
1603     $(CTFMERGE) -t -f -L VERSION -o $@ $(ZLIB_OBJS:%=$(OBJS_DIR)/%)

1605 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/hxge/%.c
1606     $(COMPILE.c) -o $@ $<
1607     $(CTFCONVERT_O)

1609 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/tpm/%.c
1610     $(COMPILE.c) -o $@ $<
1611     $(CTFCONVERT_O)

1613 $(OBJS_DIR)/%.o:      $(UTSBASE)/common/io/tpm/%.s
1614     $(COMPILE.s) -o $@ $<

1616 $(OBJS_DIR)/bz2%.o:      $(COMMONBASE)/bzip2/%.c
1617     $(COMPILE.c) -o $@ -I$(COMMONBASE)/bzip2 $<
1618     $(CTFCONVERT_O)

1620 BZ2LINT = -erroff=%all -I$(UTSBASE)/common/bzip2

1622 $(LINTS_DIR)/bz2%.ln:    $(COMMONBASE)/bzip2/%.c
1623     @$(LHEAD) $(LINT.c) -C $(LINTS_DIR)/`basename $@ .ln` $(BZ2LINT) $< $(

1625 #
1626 #      Section lb:      Lint `objects'
1627 #
1628 $(LINTS_DIR)/%.ln:      $(COMMONBASE)/crypto/aes/%.c
1629     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1631 $(LINTS_DIR)/%.ln:      $(COMMONBASE)/crypto/arcfour/%.c
1632     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1634 $(LINTS_DIR)/%.ln:      $(COMMONBASE)/crypto/blowfish/%.c
1635     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

1637 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/crypto/ecc/%.c
1638     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1640 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/crypto/modes/%.c
1641     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1643 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/crypto/padding/%.c
1644     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1646 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/crypto/rng/%.c
1647     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1649 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/crypto/rsa/%.c
1650     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1652 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/bignum/%.c
1653     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1655 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/bignum/%.c
1656     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1658 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/mpi/%.c
1659     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1661 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/acl/%.c
1662     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1664 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/avl/%.c
1665     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1667 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/ucode/%.c
1668     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1670 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/brand/snl/%.c
1671     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1673 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/brand/solaris10/%.c
1674     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1676 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/c2/%.c
1677     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1679 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/conf/%.c
1680     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1682 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/contract/%.c
1683     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1685 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/cpr/%.c
1686     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1688 $(LINTS_DIR)/%.ln:    $(UTSBASE)/common/ctf/%.c
1689     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1691 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/ctf/%.c
1692     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1694 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/pci/%.c
1695     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1697 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/devid/%.c
1698     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1700 $(LINTS_DIR)/%.ln:    $(COMMONBASE)/crypto/des/%.c
1701     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

1703 $(LINTSDIR)/%.ln:      $(COMMONBASE)/secflags/%.c
1704     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1706 $(LINTSDIR)/%.ln:      $(COMMONBASE)/smbios/%.c
1707     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1709 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/des/%.c
1710     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1712 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/crypto/api/%.c
1713     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1715 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/crypto/core/%.c
1716     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1718 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/crypto/io/%.c
1719     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1721 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/crypto/spi/%.c
1722     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1724 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/disp/%.c
1725     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1727 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/dtrace/%.c
1728     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1730 $(LINTSDIR)/%.ln:      $(COMMONBASE)/exacct/%.c
1731     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1733 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/exec/aout/%.c
1734     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1736 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/exec/elf/%.c
1737     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1739 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/exec/intp/%.c
1740     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1742 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/exec/shbin/%.c
1743     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1745 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/exec/java/%.c
1746     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1748 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/%.c
1749     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1751 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/autofs/%.c
1752     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1754 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/bootfs/%.c
1755     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1757 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/ctfs/%.c
1758     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1760 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/doorfs/%.c
1761     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1763 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/dcfs/%.c
1764     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1766 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/devfs/%.c
1767     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

1769 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/dev/%.c
1770     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1772 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/fd/%.c
1773     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1775 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/fifofs/%.c
1776     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1778 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/hsfs/%.c
1779     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1781 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/lofs/%.c
1782     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1784 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/mntfs/%.c
1785     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1787 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/namefs/%.c
1788     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1790 $(LINTSDIR)/%.ln:      $(COMMONBASE)/smbsrv/%.c
1791     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1793 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/smbsrv/%.c
1794     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1796 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/nfs/%.c
1797     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1799 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/objfs/%.c
1800     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1802 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/pcfs/%.c
1803     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1805 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/portfs/%.c
1806     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1808 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/proc/%.c
1809     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1811 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/sharefs/%.c
1812     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1814 $(LINTSDIR)/%.ln:      $(COMMONBASE)/smbclnt/%.c
1815     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1817 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/smbclnt/netmb/%.c
1818     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1820 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/smbclnt/smbfs/%.c
1821     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1823 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/socketfs/%.c
1824     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1826 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/specfs/%.c
1827     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1829 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/swapfs/%.c
1830     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1832 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/tmpfs/%.c
1833     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

1835 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/udfs/%.c
1836     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1838 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/ufs/%.c
1839     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1841 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/ufs_log/%.c
1842     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1844 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/vscan/%.c
1845     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1847 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/zfs/%.c
1848     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1850 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/zfs/luar/%.c
1851     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1853 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/fs/zut/%.c
1854     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1856 $(LINTSDIR)/%.ln:      $(COMMONBASE)/xattr/%.c
1857     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1859 $(LINTSDIR)/%.ln:      $(COMMONBASE)/zfs/%.c
1860     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1862 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/gssapi/%.c
1863     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1865 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/gssapi/mechs/dummy/%.c
1866     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1868 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/%.c
1869     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1871 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/%.c
1872     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1874 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/des/%.c
1875     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1877 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/dk/%.c
1878     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1880 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/os/%.c
1881     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1883 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/arcfour/%.c
1884     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1886 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/enc_provider/%.c
1887     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1889 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/hash_provider/%.c
1890     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1892 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/keyhash_provider/%.c
1893     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1895 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/raw/%.c
1896     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1898 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/crypto/old/%.c
1899     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

```

```

1901 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/krb5/krb/%.c
1902     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1904 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/krb5/os/%.c
1905     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1907 $(LINTSDIR)/%.ln:      $(KMECHKRB5_BASE)/mech/%.c
1908     @$(LHEAD) $(LINT.c) $(KGSSDFLAGS) $< $(LTAIL))

1910 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/idmap/%.c
1911     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1913 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/%.c
1914     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1916 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/sockmods/%.c
1917     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1919 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/arp/%.c
1920     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1922 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/ip/%.c
1923     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1925 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/ipnet/%.c
1926     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1928 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/iptun/%.c
1929     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1931 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/ipd/%.c
1932     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1934 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/ipf/%.c
1935     @$(LHEAD) $(LINT.c) $(IPFFLAGS) $< $(LTAIL))

1937 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/kssl/%.c
1938     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1940 $(LINTSDIR)/%.ln:      $(COMMONBASE)/net/patricia/%.c
1941     @$(LHEAD) $(LINT.c) $(IPFFLAGS) $< $(LTAIL))

1943 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/udp/%.c
1944     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1946 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/sctp/%.c
1947     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1949 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/tcp/%.c
1950     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1952 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/ilb/%.c
1953     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1955 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/nca/%.c
1956     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1958 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/dlpistub/%.c
1959     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1961 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/%.c
1962     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1964 $(LINTSDIR)/%.ln:      $(SRC)/common/font/%.c
1965     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

1967 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/1394/%.c
1968     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1970 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/1394/adapters/%.c
1971     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1973 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/1394/targets/av1394/%.c
1974     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1976 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/1394/targets/dcam1394/%.c
1977     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1979 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/1394/targets/scsal394/%.c
1980     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1982 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/sbp2/%.c
1983     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1985 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/aac/%.c
1986     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1988 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/afe/%.c
1989     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1991 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/atge/%.c
1992     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1994 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/arn/%.c
1995     @$(LHEAD) $(LINT.c) $< $(LTAIL))

1997 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ath/%.c
1998     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2000 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/atu/%.c
2001     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2003 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/impl/%.c
2004     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2006 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/ac97/%.c
2007     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2009 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audio1575/%.c
2010     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2012 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audio810/%.c
2013     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2015 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiocmi/%.c
2016     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2018 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiocmihd/%.c
2019     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2021 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audioens/%.c
2022     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2024 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audioemu10k/%.c
2025     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2027 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiohd/%.c
2028     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2030 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audioixp/%.c
2031     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2033 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiols/%.c
2034     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2036 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiopci/%.c
2037     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2039 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiopl6x/%.c
2040     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2042 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiosolo/%.c
2043     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2045 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiot/%.c
2046     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2048 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/audio/drv/audiovia823x/%.c
2049     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2051 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bfe/%.c
2052     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2054 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bpf/%.c
2055     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2057 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bge/%.c
2058     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2060 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/blkdev/%.c
2061     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2063 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/%.c
2064     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2066 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/577xx/common/%.c
2067     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2069 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/e
2070     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2072 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
2073     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2075 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
2076     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2078 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
2079     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2081 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
2082     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2084 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/bnxe/577xx/drivers/common/l
2085     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2087 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/cardbus/%.c
2088     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2090 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/lu/stmf_sbd/%.c
2091     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2093 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/port/fct/%.c
2094     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2096 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/port/qlt/%.c
2097     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2099 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/port/srpt/%.c
2100     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2102 $(LINTSDIR)/%.ln: $(COMMONBASE)/iscsit/%.c
2103     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2105 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/port/fcoet/%.c
2106     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2108 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/port/iscsit/%.c
2109     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2111 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/port/pppt/%.c
2112     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2114 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/comstar/stmf/%.c
2115     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2117 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/cpgary3/%.c
2118     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2120 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/dld/%.c
2121     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2123 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/dls/%.c
2124     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2126 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/dmfe/%.c
2127     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2129 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/drm/%.c
2130     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2132 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/efe/%.c
2133     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2135 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/elx1/%.c
2136     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2138 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fcoe/%.c
2139     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2141 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/hme/%.c
2142     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2144 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/pciex/%.c
2145     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2147 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/hotplug/hpcsvc/%.c
2148     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2150 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/pciex/hotplug/%.c
2151     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2153 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/hotplug/pcihp/%.c
2154     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2156 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/rds/%.c
2157     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2159 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/rdsv3/%.c
2160     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2162 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/iser/%.c
2163     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2165 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/ibd/%.c
2166     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2168 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/eoib/%.c
2169     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2171 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/of/sol_ofs/%.c
2172     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2174 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/of/sol_ucma/%.c
2175     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2177 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/of/sol_umad/%.c
2178     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2180 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/of/sol_uverbs/%.c
2181     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2183 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/sdp/%.c
2184     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2186 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/mgt/ibcm/%.c
2187     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2189 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/mgt/ibdm/%.c
2190     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2192 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/mgt/ibdma/%.c
2193     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2195 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/mgt/ibmf/%.c
2196     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2198 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/ibnex/%.c
2199     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2201 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/ibt1/%.c
2202     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2204 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/adapters/tavor/%.c
2205     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2207 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/adapters/hermon/%.c
2208     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2210 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ib/clients/daplt/%.c
2211     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2213 $(LINTSDIR)/%.ln: $(COMMONBASE)/iscsi/%.c
2214     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2216 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/idm/%.c
2217     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2219 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ipw/%.c
2220     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2222 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/iwh/%.c
2223     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2225 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/iwi/%.c
2226     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2228 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/iwk/%.c
2229     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2231 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/iwp/%.c
2232     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2234 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/kb8042/%.c
2235     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2237 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/kbtrans/%.c
2238     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2240 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ksocket/%.c
2241     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2243 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/aggr/%.c
2244     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2246 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/lp/%.c
2247     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2249 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mac/%.c
2250     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2252 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mac/plugins/%.c
2253     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2255 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mega_sas/%.c
2256     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2258 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mii/%.c
2259     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2261 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mr_sas/%.c
2262     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2264 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/scsi/adapters/mpt_sas/%.c
2265     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2267 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mxfe/%.c
2268     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2270 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mwl/%.c
2271     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2273 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/mwl/mwl_fw/%.c
2274     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2276 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/net80211/%.c
2277     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2279 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/nge/%.c
2280     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2282 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/nvme/%.c
2283     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2285 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/nxge/%.c
2286     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2288 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/nxge/%.s
2289     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2291 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/nxge/npi/%.c
2292     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2294 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/pci-ide/%.c
2295     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2297 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/pcn/%.c
2298     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2300 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ppp/sppp/%.c
2301     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2303 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ppp/sppasyn/%.c
2304     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2306 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ppp/sppptun/%.c
2307     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2309 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/qede/%.c
2310     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2312 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ral/%.c
2313     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2315 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/rge/%.c
2316     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2318 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/rtls/%.c
2319     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2321 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/rsm/%.c
2322     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2324 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/rtw/%.c
2325     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2327 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/rum/%.c
2328     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2330 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/rwd/%.c
2331     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2333 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/rwn/%.c
2334     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2336 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/sata/adapters/ahci/%.c
2337     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2339 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/sata/adapters/nv_sata/%.c
2340     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2342 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/sata/adapters/si3124/%.c
2343     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2345 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/sata/impl/%.c
2346     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2348 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/scsi/adapters/%.c
2349     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2351 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/scsi/adapters/blk2scsa/%.c
2352     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2354 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/scsi/adapters/pmcs/%.c
2355     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2357 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/scsi/adapters/scsi_vhci/%.c
2358     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2360 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/scsi/adapters/scsi_vhci/fop
2361     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2363 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fibre-channel/ulp/%.c
2364     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2366 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fibre-channel/impl/%.c
2367     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2369 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fibre-channel/fca/qlc/%.c
2370     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2372 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fibre-channel/fca/qlge/%.c
2373     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2375 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fibre-channel/fca/emlxs/%.c
2376     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2378 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fibre-channel/fca/oce/%.c
2379     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2381 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/fibre-channel/fca/fcoei/%.c
2382     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2384 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/scsi/conf/%.c
2385     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2387 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/scsi/impl/%.c
2388     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2390 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/scsi/targets/%.c
2391     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2393 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/sdcard/adapters/sdhost/%.c
2394     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2396 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/sdcard/impl/%.c
2397     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2399 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/sdcard/targets/sdcard/%.c
2400     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2402 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/sfe/%.c
2403     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2405 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/simnet/%.c
2406     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2408 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/softmac/%.c
2409     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2411 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/uath/%.c
2412     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2414 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/uath/uath_fw/%.c
2415     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2417 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/ural/%.c
2418     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2420 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/urtw/%.c
2421     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2423 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/audio/usb_ac/%.
2424     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2426 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/audio/usb_as/%.
2427     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2429 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/audio/usb_ah/%.
2430     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2432 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbskel/%.c
2433     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2435 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/video/usbvc/%.c
2436     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2438 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/hid/%.c
2439     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2441 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/hidparser/%.c
2442     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2444 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbkbm/%.c
2445     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2447 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbms/%.c
2448     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2450 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbinput/usbwcm
2451     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2453 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/ugen/%.c
2454     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2456 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/printer/%.c
2457     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2459 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbser/%.c
2460     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2462 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbser/usbsacm/
2463     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2465 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbser/usbftdi/
2466     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2468 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbser/usbser_k
2469     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2471 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbser/usbsprl/
2472     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2474 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/clients/usbecm/%.c
2475     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2477 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/hcd/openhci/%.c
2478     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2480 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/hcd/ehci/%.c
2481     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2483 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/hcd/xhci/%.c
2484     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2486 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/hcd/uhci/%.c
2487     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2489 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/hubd/%.c
2490     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2492 $(LINTSDIR)/%.ln: $(UTSBASE)/common/io/usb/scsa2usb/%.c
2493     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2495 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/usb/usb_mid/%.c
2496     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2498 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/usb/usb_ia/%.c
2499     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2501 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/usb/usba/%.c
2502     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2504 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/usb/usba10/%.c
2505     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2507 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/usbgem/%.c
2508     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2510 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/axf/%.c
2511     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2513 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/udmf/%.c
2514     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2516 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/upf/%.c
2517     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2519 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/urf/%.c
2520     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2522 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/vuidmice/%.c
2523     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2525 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/vnic/%.c
2526     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2528 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/wpi/%.c
2529     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2531 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/zyd/%.c
2532     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2534 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/chxge/com/%.c
2535     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2537 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/chxge/%.c
2538     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2540 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/cxgbe/common/%.c
2541     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2543 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/cxgbe/shared/%.c
2544     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2546 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/cxgbe/firmware/%.c
2547     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2549 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/cxgbe/t4nex/%.c
2550     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2552 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/cxgbe/cxgbe/%.c
2553     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2555 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ixgb/%.c
2556     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2558 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/xge/drv/%.c
2559     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2561 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/xge/hal/xgehal/%.c
2562     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2564 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/e1000g/%.c
2565     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2567 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/e1000api/%.c
2568     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2570 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/igb/%.c
2571     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2573 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/iprb/%.c
2574     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2576 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ixgbe/%.c
2577     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2579 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ixgbe/core/%.c
2580     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2582 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/i40e/%.c
2583     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2585 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/ntxn/%.c
2586     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2588 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/myril0ge/drv/%.c
2589     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2591 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/ipp/%.c
2592     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2594 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/ipp/ipgpc/%.c
2595     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2597 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/ipp/dlcosmk/%.c
2598     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2600 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/ipp/flowacct/%.c
2601     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2603 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/ipp/dscpmk/%.c
2604     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2606 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/ipp/meters/%.c
2607     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2609 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/kiconv/kiconv_emea/%.c
2610     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2612 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/kiconv/kiconv_ja/%.c
2613     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2615 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/kiconv/kiconv_ko/%.c
2616     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2618 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/kiconv/kiconv_sc/%.c
2619     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2621 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/kiconv/kiconv_tc/%.c
2622     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2624 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/klm/%.c
2625     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```



```

2627 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/kmdb/%.c
2628     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2630 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/krtld/%.c
2631     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2633 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/ktli/%.c
2634     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2636 $(LINTSDIR)/%.ln:      $(COMMONBASE)/list/%.c
2637     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2639 $(LINTSDIR)/%.ln:      $(COMMONBASE)/lzma/%.c
2640     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2642 $(LINTSDIR)/%.ln:      $(COMMONBASE)/crypto/md4/%.c
2643     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2645 $(LINTSDIR)/%.ln:      $(COMMONBASE)/crypto/md5/%.c
2646     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2648 $(LINTSDIR)/%.ln:      $(COMMONBASE)/net/dhcp/%.c
2649     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2651 $(LINTSDIR)/%.ln:      $(COMMONBASE)/nvpair/%.c
2652     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2654 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/os/%.c
2655     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2657 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/rpc/%.c
2658     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2660 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/pcmcia/cs/%.c
2661     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2663 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/pcmcia/cis/%.c
2664     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2666 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/pcmcia/nexus/%.c
2667     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2669 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/pcmcia/pcs/%.c
2670     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2672 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/rpc/%.c
2673     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2675 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/rpc/sec/%.c
2676     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2678 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/rpc/sec_gss/%.c
2679     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2681 $(LINTSDIR)/%.ln:      $(COMMONBASE)/crypto/edonr/%.c
2682     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2684 $(LINTSDIR)/%.ln:      $(COMMONBASE)/crypto/shal/%.c
2685     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2687 $(LINTSDIR)/%.ln:      $(COMMONBASE)/crypto/sha2/%.c
2688     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2690 $(LINTSDIR)/%.ln:      $(COMMONBASE)/crypto/skein/%.c
2691     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

2693 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/syscall/%.c
2694     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2696 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/tnf/%.c
2697     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2699 $(LINTSDIR)/%.ln:      $(COMMONBASE)/tsol/%.c
2700     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2702 $(LINTSDIR)/%.ln:      $(COMMONBASE)/util/%.c
2703     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2705 $(LINTSDIR)/%.ln:      $(COMMONBASE)/unicode/%.c
2706     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2708 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/vm/%.c
2709     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2711 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/scsi/adapters/iscsi/%.c
2712     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2714 $(LINTSDIR)/%.ln:      $(COMMONBASE)/iscsi/%.c
2715     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2717 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/inet/kifconf/%.c
2718     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2720 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/virtio/%.c
2721     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2723 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/vioif/%.c
2724     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2726 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/vioif/%.c
2727     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2729 ZMODLINTFLAGS = -erroff=E_CONSTANT_CONDITION

2731 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/zmod/%.c
2732     @$(LHEAD) $(LINT.c) $(ZMODLINTFLAGS) $< $(LTAIL))

2734 $(LINTSDIR)/zlib_obj.ln: $(ZLIB_OBJS:%.o=$(LINTSDIR)/%.ln) \
2735     $(UTSBASE)/common/zmod/zlib_lint.c
2736     @$(LHEAD) $(LINT.c) -C $(LINTSDIR)/zlib_obj \
2737     $(UTSBASE)/common/zmod/zlib_lint.c $(LTAIL))

2739 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/hxge/%.c
2740     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2742 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/tpm/%.c
2743     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2745 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/tpm/%.s
2746     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2748 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/vr/%.c
2749     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2751 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/yge/%.c
2752     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2754 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/sfxge/%.c
2755     @$(LHEAD) $(LINT.c) $< $(LTAIL))

2757 $(LINTSDIR)/%.ln:      $(UTSBASE)/common/io/sfxge/common/%.c

```

```
2758      @$(LHEAD) $(LINT.c) $< $(LTAIL))
2760 $(LINTS_DIR)/%.ln:      $(UTSBASE)/common/io/skd/%.c
2761      @$(LHEAD) $(LINT.c) $< $(LTAIL))
2763 $(LINTS_DIR)/%.ln:      $(COMMONBASE)/fsreparse/%.c
2764      @$(LHEAD) $(LINT.c) $< $(LTAIL))
```

new/usr/src/uts/common/sys/ctf_api.h

1

```
*****
14030 Thu Apr 25 02:20:49 2019
new/usr/src/uts/common/sys/ctf_api.h
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License").  You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23  * Copyright 2005 Sun Microsystems, Inc.  All rights reserved.
24  * Use is subject to license terms.
25 */
26 /*
27  * Copyright 2019 Joyent, Inc.
28  * Copyright 2018 Joyent, Inc.
29 */
30 /*
31  * This header file defines the interfaces available from the CTF debugger
32  * library, libctf, and an equivalent kernel module.  This API can be used by
33  * a debugger to operate on data in the Compact ANSI-C Type Format (CTF).
34  * This is NOT a public interface, although it may eventually become one in
35  * the fullness of time after we gain more experience with the interfaces.
36  *
37  * In the meantime, be aware that any program linked with this API in this
38  * release of Solaris is almost guaranteed to break in the next release.
39  *
40  * In short, do not user this header file or the CTF routines for any purpose.
41  */
42
43 #ifndef _CTF_API_H
44 #define _CTF_API_H
45
46 #include <sys/types.h>
47 #include <sys/param.h>
48 #include <sys/elf.h>
49 #include <sys/ctf.h>
50
51 #ifdef __cplusplus
52 extern "C" {
```

new/usr/src/uts/common/sys/ctf_api.h

2

```
53 #endif
54
55 /*
56  * Clients can open one or more CTF containers and obtain a pointer to an
57  * opaque ctf_file_t.  Types are identified by an opaque ctf_id_t token.
58  * These opaque definitions allow libctf to evolve without breaking clients.
59  */
60 typedef struct ctf_file ctf_file_t;
61 typedef long ctf_id_t;
62
63 #define ECTF_BASE 1000 /* base value for libctf errnos */
64
65 enum {
66     ECTF_FMT = ECTF_BASE, /* file is not in CTF or ELF format */
67     ECTF_ELFVERS, /* ELF version is more recent than libctf */
68     ECTF_CTFVERS, /* CTF version is more recent than libctf */
69     ECTF_ENDIAN, /* data is different endian-ness than lib */
70     ECTF_SYMTAB, /* symbol table uses invalid entry size */
71     ECTF_SYMBAD, /* symbol table data buffer invalid */
72     ECTF_STRBAD, /* string table data buffer invalid */
73     ECTF_CORRUPT, /* file data corruption detected */
74     ECTF_NOCTFDATA, /* ELF file does not contain CTF data */
75     ECTF_NOCTFBUF, /* buffer does not contain CTF data */
76     ECTF_NOSYMTAB, /* symbol table data is not available */
77     ECTF_NOPARENT, /* parent CTF container is not available */
78     ECTF_DMODEL, /* data model mismatch */
79     ECTF_MMAP, /* failed to mmap a data section */
80     ECTF_ZMISSING, /* decompression library not installed */
81     ECTF_ZINIT, /* failed to initialize decompression library */
82     ECTF_ZALLOC, /* failed to allocate decompression buffer */
83     ECTF_DECOMPRESS, /* failed to decompress CTF data */
84     ECTF_STRTAB, /* string table for this string is missing */
85     ECTF_BADNAME, /* string offset is corrupt w.r.t. strtabs */
86     ECTF_BADID, /* invalid type ID number */
87     ECTF_NOTSOU, /* type is not a struct or union */
88     ECTF_NOTENUM, /* type is not an enum */
89     ECTF_NOTSUE, /* type is not a struct, union, or enum */
90     ECTF_NOTINTFP, /* type is not an integer or float */
91     ECTF_NOTARRAY, /* type is not an array */
92     ECTF_NOTREF, /* type does not reference another type */
93     ECTF_NAMELEN, /* buffer is too small to hold type name */
94     ECTF_NOTYPE, /* no type found corresponding to name */
95     ECTF_SYNTAX, /* syntax error in type name */
96     ECTF_NOTFUNC, /* symtab entry does not refer to a function */
97     ECTF_NOFUNCDAT, /* no func info available for function */
98     ECTF_NOTDATA, /* symtab entry does not refer to a data obj */
99     ECTF_NOTYPEDAT, /* no type info available for object */
100    ECTF_NOLABEL, /* no label found corresponding to name */
101    ECTF_NOLABELDATA, /* file does not contain any labels */
102    ECTF_NOTSUP, /* feature not supported */
103    ECTF_NOENUMNAM, /* enum element name not found */
104    ECTF_NOMEMBNAM, /* member name not found */
105    ECTF_RDONLY, /* CTF container is read-only */
106    ECTF_DTFULL, /* CTF type is full (no more members allowed) */
107    ECTF_FULL, /* CTF container is full */
108    ECTF_DUPMEMBER, /* duplicate member name definition */
109    ECTF_CONFLICT, /* conflicting type definition present */
110    ECTF_REFERENCED, /* type has outstanding references */
111    ECTF_NOTDYN, /* type is not a dynamic type */
112    ECTF_ELF, /* elf library failure */
113    ECTF_MCHILD, /* cannot merge child container */
114    ECTF_LABEL EXISTS, /* label already exists */
115    ECTF_LCONFLICT, /* merged labels conflict */
116    ECTF_ZLIB, /* zlib library failure */
117    ECTF_CONVBACKERR, /* CTF conversion backend error */
118    ECTF_CONVNOCSRC, /* No C source to convert from */
```

new/usr/src/uts/common/sys/ctf_api.h

3

```
119     ECTF_CONVNODEBUG      /* No debug info to convert into CTF */
119     ECTF_NOCONVBKEND     /* No applicable conversion backend */
120 };
_____unchanged_portion_omitted_
```

```

*****
13511 Thu Apr 25 02:20:49 2019
new/usr/src/uts/intel/Makefile.rules
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 # Copyright 2019 Joyent, Inc. All rights reserved.
25 # Copyright 2018 Joyent, Inc. All rights reserved.
26 # Copyright 2017 Nexenta Systems, Inc.
27 #
28 #
29 # This Makefile defines all file modules and build rules for the
30 # directory uts/intel and its children. These are the source files which
31 # are specific to the intel processor.
32 #
33 # The following two-level ordering must be maintained in this file.
34 # Lines are sorted first in order of decreasing specificity based on
35 # the first directory component. That is, sun4u rules come before
36 # sparc rules come before common rules.
37 #
38 # Lines whose initial directory components are equal are sorted
39 # alphabetically by the remaining components.
40 #
41 #
42 # Need a way to distinguish between the ia32 and amd64 subdirs.
43 #
44 SUBARCH_DIR_32 = ia32
45 SUBARCH_DIR_64 = amd64
46 SUBARCH_DIR = ${SUBARCH_DIR_$(CLASS)}
47 #
48 #
49 # Section 1a: C object build rules
50 #
51 $(OBJSDIR)/%.o: $(SRC)/common/fs/%.c
52 $(COMPILE.c) -o $@ $<

```

```

53 $(CTFCONVERT_O)
54 #
55 $(OBJSDIR)/%.o: $(UTSBASE)/common/io/power/%.c
56 $(COMPILE.c) -o $@ $<
57 $(CTFCONVERT_O)
58 #
59 $(OBJSDIR)/%.o: $(SRC)/common/util/i386/%.s
60 $(COMPILE.s) -o $@ $<
61 #
62 $(OBJSDIR)/%.o: $(UTSBASE)/intel/brand/sn1/%.s
63 $(COMPILE.s) -o $@ $<
64 #
65 $(OBJSDIR)/%.o: $(UTSBASE)/intel/brand/solaris10/%.s
66 $(COMPILE.s) -o $@ $<
67 #
68 $(OBJSDIR)/%.o: $(UTSBASE)/intel/dtrace/%.c
69 $(COMPILE.c) -o $@ $<
70 $(CTFCONVERT_O)
71 #
72 $(OBJSDIR)/%.o: $(UTSBASE)/intel/dtrace/%.s
73 $(COMPILE.s) -o $@ $<
74 #
75 $(OBJSDIR)/%.o: $(UTSBASE)/intel/fs/proc/%.c
76 $(COMPILE.c) -o $@ $<
77 $(CTFCONVERT_O)
78 #
79 $(OBJSDIR)/%.o: $(UTSBASE)/intel/ia32/ml/%.s
80 $(COMPILE.s) -o $@ $<
81 #
82 $(OBJSDIR)/%.o: $(UTSBASE)/intel/ia32/os/%.c
83 $(COMPILE.c) -o $@ $<
84 $(CTFCONVERT_O)
85 #
86 $(OBJSDIR)/%.o: $(UTSBASE)/intel/ia32/promif/%.c
87 $(COMPILE.c) -o $@ $<
88 $(CTFCONVERT_O)
89 #
90 $(OBJSDIR)/%.o: $(UTSBASE)/intel/ia32/syscall/%.c
91 $(COMPILE.c) -o $@ $<
92 $(CTFCONVERT_O)
93 #
94 $(OBJSDIR)/%.o: $(UTSBASE)/intel/io/%.c
95 $(COMPILE.c) -o $@ $<
96 $(CTFCONVERT_O)
97 #
98 $(OBJSDIR)/%.o: $(UTSBASE)/intel/io/acpica/%.c
99 $(COMPILE.c) -o $@ $<
100 $(CTFCONVERT_O)
101 #
102 $(OBJSDIR)/%.o: $(UTSBASE)/intel/io/acpica/%.s
103 $(COMPILE.s) -o $@ $<
104 #
105 $(OBJSDIR)/%.o: $(SRC)/common/acpica/events/%.c
106 $(COMPILE.c) -o $@ $<
107 $(CTFCONVERT_O)
108 #
109 $(OBJSDIR)/%.o: $(SRC)/common/acpica/hardware/%.c
110 $(COMPILE.c) -o $@ $<
111 $(CTFCONVERT_O)
112 #
113 $(OBJSDIR)/%.o: $(SRC)/common/acpica/dispatcher/%.c
114 $(COMPILE.c) -o $@ $<
115 $(CTFCONVERT_O)
116 #
117 $(OBJSDIR)/%.o: $(SRC)/common/acpica/executer/%.c
118 $(COMPILE.c) -o $@ $<

```

```

119      $(CTFCONVERT_O)

121 $(OBJS_DIR)/%.o:      $(SRC)/common/acpica/parser/%.c
122   $(COMPILE.c) -o $@ $<
123   $(CTFCONVERT_O)

125 $(OBJS_DIR)/%.o:      $(SRC)/common/acpica/namespace/%.c
126   $(COMPILE.c) -o $@ $<
127   $(CTFCONVERT_O)

129 $(OBJS_DIR)/%.o:      $(SRC)/common/acpica/resources/%.c
130   $(COMPILE.c) -o $@ $<
131   $(CTFCONVERT_O)

133 $(OBJS_DIR)/%.o:      $(SRC)/common/acpica/tables/%.c
134   $(COMPILE.c) -o $@ $<
135   $(CTFCONVERT_O)

137 $(OBJS_DIR)/%.o:      $(SRC)/common/acpica/utilities/%.c
138   $(COMPILE.c) -o $@ $<
139   $(CTFCONVERT_O)

141 $(OBJS_DIR)/%.o:      $(SRC)/common/acpica/disassembler/%.c
142   $(COMPILE.c) -o $@ $<
143   $(CTFCONVERT_O)

145 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/agpgart/%.c
146   $(COMPILE.c) -o $@ $<
147   $(CTFCONVERT_O)

149 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/agpmaster/%.c
150   $(COMPILE.c) -o $@ $<
151   $(CTFCONVERT_O)

153 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/amd8111s/%.c
154   $(COMPILE.c) -o $@ $<
155   $(CTFCONVERT_O)

157 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/amr/%.c
158   $(COMPILE.c) -o $@ $<
159   $(CTFCONVERT_O)

161 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/drm/%.c
162   $(COMPILE.c) -o $@ $<
163   $(CTFCONVERT_O)

165 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/hotplug/pcicfg/%.c
166   $(COMPILE.c) -o $@ $<
167   $(CTFCONVERT_O)

169 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/ipmi/%.c
170   $(COMPILE.c) -o $@ $<
171   $(CTFCONVERT_O)

173 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/intel_nb5000/%.c
174   $(COMPILE.c) -o $@ $<
175   $(CTFCONVERT_O)

177 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/intel_nhm/%.c
178   $(COMPILE.c) -o $@ $<
179   $(CTFCONVERT_O)

181 $(OBJS_DIR)/%.o:      $(SRC)/common/mc/mc-amd/%.c
182   $(COMPILE.c) -o $@ $<
183   $(CTFCONVERT_O)

```

```

185 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/mc-amd/%.c
186   $(COMPILE.c) -o $@ $<
187   $(CTFCONVERT_O)

189 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/pci/%.c
190   $(COMPILE.c) -o $@ $<
191   $(CTFCONVERT_O)

193 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/pciex/%.c
194   $(COMPILE.c) -o $@ $<
195   $(CTFCONVERT_O)

197 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/dktp/controller/ata/%.c
198   $(COMPILE.c) -o $@ $<
199   $(CTFCONVERT_O)

201 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/dktp/dcdev/%.c
202   $(COMPILE.c) -o $@ $<
203   $(CTFCONVERT_O)

205 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/dktp/disk/%.c
206   $(COMPILE.c) -o $@ $<
207   $(CTFCONVERT_O)

209 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/dktp/drvojb/%.c
210   $(COMPILE.c) -o $@ $<
211   $(CTFCONVERT_O)

213 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/dktp/hba/ghd/%.c
214   $(COMPILE.c) -o $@ $<
215   $(CTFCONVERT_O)

217 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/dnet/%.c
218   $(COMPILE.c) -o $@ $<
219   $(CTFCONVERT_O)

221 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/scsi/adapters/arcmsr/%.c
222   $(COMPILE.c) -o $@ $<
223   $(CTFCONVERT_O)

225 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/scsi/targets/%.c
226   $(COMPILE.c) -o $@ $<
227   $(CTFCONVERT_O)

229 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/vgatext/%.c
230   $(COMPILE.c) -o $@ $<
231   $(CTFCONVERT_O)

233 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/vmxnet3s/%.c
234   $(COMPILE.c) -o $@ $<
235   $(CTFCONVERT_O)

237 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/io/scsi/adapters/pvscsi/%.c
238   $(COMPILE.c) -o $@ $<
239   $(CTFCONVERT_O)

241 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/nskern/%.s
242   $(COMPILE.s) -o $@ $<

244 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/os/%.c
245   $(COMPILE.c) -o $@ $<
246   $(CTFCONVERT_O)

248 $(OBJS_DIR)/%.o:      $(UTSBASE)/intel/pcbe/%.c
249   $(COMPILE.c) -o $@ $<
250   $(CTFCONVERT_O)

```

```

252 $(OBJDIR)/%.o: $(UTSBASE)/intel/promif/%.c
253 $(COMPILE.c) -o $@ $<
254 $(CTFCONVERT_O)

256 $(OBJDIR)/%.o: $(UTSBASE)/intel/syscall/%.c
257 $(COMPILE.c) -o $@ $<
258 $(CTFCONVERT_O)

260 $(OBJDIR)/%.o: $(UTSBASE)/common/os/%.c
261 $(COMPILE.c) -o $@ $<
262 $(CTFCONVERT_O)

264 $(OBJDIR)/%.o: $(UTSBASE)/intel/kdi/%.c
265 $(COMPILE.c) -o $@ $<
266 $(CTFCONVERT_O)

268 $(OBJDIR)/%.o: $(UTSBASE)/intel/kdi/%.s
269 $(COMPILE.s) -o $@ $<

271 $(OBJDIR)/%.o: $(UTSBASE)/intel/zfs/%.c
272 $(COMPILE.c) -o $@ $<
273 $(CTFCONVERT_O)

275 #
276 # krtld compiled into unix
277 #

279 KRTLD_INC_PATH = -I$(UTSBASE)/common/krtld -I$(UTSBASE)/intel/sys
280 KRTLD_INC_PATH += -I$(UTSBASE)/intel/$(SUBARCH_DIR)/krtld

282 KRTLD_CPPFLAGS_32 = -DELFTARGET_386
283 KRTLD_CPPFLAGS_64 = -DELFTARGET_AMD64 -DMODDIR_SUFFIX="\amd64\"
284 KRTLD_CPPFLAGS = $(KRTLD_CPPFLAGS_$(CLASS)) -D_KRTLD

286 $(OBJDIR)/%.o: $(UTSBASE)/common/krtld/%.c
287 $(COMPILE.c) $(KRTLD_INC_PATH) $(KRTLD_CPPFLAGS) -o $@ $<
288 $(CTFCONVERT_O)

290 $(OBJDIR)/%.o: $(UTSBASE)/intel/$(SUBARCH_DIR)/krtld/%.c
291 $(COMPILE.c) $(KRTLD_INC_PATH) $(KRTLD_CPPFLAGS) -o $@ $<
292 $(CTFCONVERT_O)

294 #
295 # _DBOOT indicates that krtld is called from a dboot ELF section
296 #
297 $(OBJDIR)/kobj.o := CPPFLAGS += -D_DBOOT

299 $(OBJDIR)/%.o: $(UTSBASE)/intel/$(SUBARCH_DIR)/krtld/%.s
300 $(COMPILE.s) $(KRTLD_INC_PATH) $(KRTLD_CPPFLAGS) -o $@ $<
301 $(CTFCONVERT_O)

302 $(OBJDIR)/%.o: $(SRC)/common/util/$(SUBARCH_DIR)/%.c
303 $(COMPILE.c) $(KRTLD_INC_PATH) $(KRTLD_CPPFLAGS) -o $@ $<
304 $(CTFCONVERT_O)

307 #
308 # Section lb: Lint 'object' build rules.
309 #
310 $(LINTSDIR)/%.ln: $(SRC)/common/fs/%.c
311 @$(LHEAD) $(LINT.c) $< $(LTAIL))

313 $(LINTSDIR)/%.ln: $(SRC)/common/util/i386/%.s
314 @$(LHEAD) $(LINT.s) $< $(LTAIL))

```

```

316 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/brand/snl/%.s
317 @$(LHEAD) $(LINT.s) $< $(LTAIL))

319 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/brand/solaris10/%.s
320 @$(LHEAD) $(LINT.s) $< $(LTAIL))

322 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/dtrace/%.c
323 @$(LHEAD) $(LINT.c) $< $(LTAIL))

325 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/dtrace/%.s
326 @$(LHEAD) $(LINT.s) $< $(LTAIL))

328 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/zfs/%.c
329 @$(LHEAD) $(LINT.c) $< $(LTAIL))

331 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/fs/proc/%.c
332 @$(LHEAD) $(LINT.c) $< $(LTAIL))

334 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/ia32/ml/%.s
335 @$(LHEAD) $(LINT.s) $< $(LTAIL))

337 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/ia32/os/%.c
338 @$(LHEAD) $(LINT.c) $< $(LTAIL))

340 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/ia32/promif/%.c
341 @$(LHEAD) $(LINT.c) $< $(LTAIL))

343 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/ia32/syscall/%.c
344 @$(LHEAD) $(LINT.c) $< $(LTAIL))

346 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/io/%.c
347 @$(LHEAD) $(LINT.c) $< $(LTAIL))

349 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/io/acpica/%.s
350 @$(LHEAD) $(LINT.s) $< $(LTAIL))

352 $(LINTSDIR)/%.ln: $(UTSBASE)/intel/io/acpica/%.c
353 @$(LHEAD) $(LINT.c) $< $(LTAIL))

355 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/events/%.c
356 @$(LHEAD) $(LINT.c) $< $(LTAIL))

358 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/hardware/%.c
359 @$(LHEAD) $(LINT.c) $< $(LTAIL))

361 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/dispatcher/%.c
362 @$(LHEAD) $(LINT.c) $< $(LTAIL))

364 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/executer/%.c
365 @$(LHEAD) $(LINT.c) $< $(LTAIL))

367 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/parser/%.c
368 @$(LHEAD) $(LINT.c) $< $(LTAIL))

370 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/namespace/%.c
371 @$(LHEAD) $(LINT.c) $< $(LTAIL))

373 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/resources/%.c
374 @$(LHEAD) $(LINT.c) $< $(LTAIL))

376 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/tables/%.c
377 @$(LHEAD) $(LINT.c) $< $(LTAIL))

379 $(LINTSDIR)/%.ln: $(SRC)/common/acpica/utilities/%.c
380 @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

382 $(LINTS_DIR)/%.ln: $(SRC)/common/acpica/disassembler/%.c
383     @$(LHEAD) $(LINT.c) $< $(LTAIL))

385 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/agpgart/%.c
386     @$(LHEAD) $(LINT.c) $< $(LTAIL))

388 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/agpmaster/%.c
389     @$(LHEAD) $(LINT.c) $< $(LTAIL))

391 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/amd8111s/%.c
392     @$(LHEAD) $(LINT.c) $< $(LTAIL))

394 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/amr/%.c
395     @$(LHEAD) $(LINT.c) $< $(LTAIL))

397 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/scsi/adapters/arcmsr/%.c
398     @$(LHEAD) $(LINT.c) $< $(LTAIL))

400 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/drm/%.c
401     @$(LHEAD) $(LINT.c) $< $(LTAIL))

403 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/hotplug/pcicfg/%.c
404     @$(LHEAD) $(LINT.c) $< $(LTAIL))

406 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/hotplug/pci/%.c
407     @$(LHEAD) $(LINT.c) $< $(LTAIL))

409 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/intel_nb5000/%.c
410     @$(LHEAD) $(LINT.c) $< $(LTAIL))

412 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/intel_nhm/%.c
413     @$(LHEAD) $(LINT.c) $< $(LTAIL))

415 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/ipmi/%.c
416     @$(LHEAD) $(LINT.c) $< $(LTAIL))

418 $(LINTS_DIR)/%.ln: $(SRC)/common/mc/mc-amd/%.c
419     @$(LHEAD) $(LINT.c) $< $(LTAIL))

421 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/mc-amd/%.c
422     @$(LHEAD) $(LINT.c) $< $(LTAIL))

424 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/pci/%.c
425     @$(LHEAD) $(LINT.c) $< $(LTAIL))

427 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/pciex/%.c
428     @$(LHEAD) $(LINT.c) $< $(LTAIL))

430 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/dktp/controller/ata/%.c
431     @$(LHEAD) $(LINT.c) $< $(LTAIL))

433 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/dktp/dcdev/%.c
434     @$(LHEAD) $(LINT.c) $< $(LTAIL))

436 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/dktp/disk/%.c
437     @$(LHEAD) $(LINT.c) $< $(LTAIL))

439 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/dktp/drvoobj/%.c
440     @$(LHEAD) $(LINT.c) $< $(LTAIL))

442 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/dktp/hba/ghd/%.c
443     @$(LHEAD) $(LINT.c) $< $(LTAIL))

445 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/dnet/%.c
446     @$(LHEAD) $(LINT.c) $< $(LTAIL))

```

```

448 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/scsi/targets/%.c
449     @$(LHEAD) $(LINT.c) $< $(LTAIL))

451 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/vgatext/%.c
452     @$(LHEAD) $(LINT.c) $< $(LTAIL))

454 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/vmxnet3s/%.c
455     @$(LHEAD) $(LINT.c) $< $(LTAIL))

457 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/io/scsi/adapters/pvscsi/%.c
458     @$(LHEAD) $(LINT.c) $< $(LTAIL))

460 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/os/%.c
461     @$(LHEAD) $(LINT.c) $< $(LTAIL))

463 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/pcbe/%.c
464     @$(LHEAD) $(LINT.c) $< $(LTAIL))

466 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/promif/%.c
467     @$(LHEAD) $(LINT.c) $< $(LTAIL))

469 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/syscall/%.c
470     @$(LHEAD) $(LINT.c) $< $(LTAIL))

472 $(LINTS_DIR)/%.ln: $(UTSBASE)/common/os/%.c
473     @$(LHEAD) $(LINT.c) $< $(LTAIL))

475 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/kdi/%.c
476     @$(LHEAD) $(LINT.c) $< $(LTAIL))

478 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/kdi/%.s
479     @$(LHEAD) $(LINT.s) $< $(LTAIL))

481 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/nskern/%.s
482     @$(LHEAD) $(LINT.s) $< $(LTAIL))

484 #
485 # krtld lints
486 #
487 $(LINTS_DIR)/%.ln: $(UTSBASE)/common/krtld/%.c
488     @$(LHEAD) $(LINT.c) $(KRTLDD_INC_PATH) $(KRTLDD_CPPFLAGS) $< $(LTAIL))

490 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/$(SUBARCH_DIR)/krtld/%.c
491     @$(LHEAD) $(LINT.c) $(KRTLDD_INC_PATH) $(KRTLDD_CPPFLAGS) $< $(LTAIL))

493 $(LINTS_DIR)/%.ln: $(UTSBASE)/intel/$(SUBARCH_DIR)/krtld/%.s
494     @$(LHEAD) $(LINT.s) $(KRTLDD_INC_PATH) $(KRTLDD_CPPFLAGS) $< $(LTAIL))

496 $(LINTS_DIR)/%.ln: $(SRC)/common/util/$(SUBARCH_DIR)/%.c
497     @$(LHEAD) $(LINT.c) $(KRTLDD_INC_PATH) $(KRTLDD_CPPFLAGS) $< $(LTAIL))

499 $(OBSJ_DIR)/kobj.ln := CPPFLAGS += -D_DBOOT

```


new/usr/src/uts/intel/aes/Makefile

1

```
*****
2492 Thu Apr 25 02:20:50 2019
new/usr/src/uts/intel/aes/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 2014 Garrett D'Amore <garrett@damore.org>
22 #
23 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 #
26 # Copyright (c) 2019, Joyent, Inc.
27 # This makefile drives the production of the AES KEF provider.
28 # Intel implementation architecture dependent
29 #
30 #
31 # Path to the base of the uts directory tree (usually /usr/src/uts).
32 UTSBASE = ../../
33 COM_DIR = $(COMMONBASE)/crypto/aes
34 #
35 #
36 # Define the module and object file sets.
37 #
38 MODULE = aes
39 LINTS = $(AESPROV_OBJS:%.o=$(LINTS_DIR)/%.ln)
40 AESPROV_OBJS_32 =
41 AESPROV_OBJS_64 = aes_amd64.o aes_intel.o aeskey.o
42 AESPROV_OBJS += $(AESPROV_OBJS_$(CLASS))
43 OBJECTS = $(AESPROV_OBJS:%=$(OBJDIR)/%)
44 ROOTMODULE = $(ROOT_CRYPTO_DIR)/$(MODULE)
45 #
46 #
47 # Include common rules.
48 #
49 include $(UTSBASE)/intel/Makefile.intel
```

new/usr/src/uts/intel/aes/Makefile

2

```
51 #
52 # Define targets
53 #
54 ALL_TARGET = $(BINARY)
55 LINT_TARGET = $(MODULE).lint
56 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)
57 #
58 #
59 # Linkage dependencies
60 #
61 LDFLAGS += -dy -Nmisc/kcf
62 #
63 CPPFLAGS += -I$(COM_DIR) -I$(COM_DIR)/..
64 CPPFLAGS += -DCRYPTO_PROVIDER_NAME=\"$(MODULE)\"
65 AS_CPPFLAGS += -I../$(PLATFORM)
66 #
67 #
68 # Default build targets.
69 #
70 .KEEP_STATE:
71 #
72 def: $(DEF_DEPS)
73 #
74 all: $(ALL_DEPS)
75 #
76 clean: $(CLEAN_DEPS)
77 #
78 clobber: $(CLOBBER_DEPS)
79 #
80 lint: $(LINT_DEPS)
81 #
82 modlintlib: $(MODLINTLIB_DEPS)
83 #
84 clean.lint: $(CLEAN_LINT_DEPS)
85 #
86 install: $(INSTALL_DEPS)
87 #
88 #
89 # Include common targets.
90 #
91 include $(UTSBASE)/intel/Makefile.targ
92 #
93 $(OBJDIR)/%.ln: $(COM_DIR)/amd64/%.c
94 @($(LHEAD) $(LINT.c) $(COM_DIR)/amd64/$(@F:.ln=.c) $(LTAIL))
95 #
96 $(OBJDIR)/%.o: $(COM_DIR)/amd64/%.c
97 $(COMPILE.c) -o $@ $(COM_DIR)/amd64/$(@F:.o=.c)
98 $(CTFCONVERT_O)
99 $(POST_PROCESS_O)
100 #
101 $(OBJDIR)/%.s: $(COM_DIR)/amd64/%.s
102 $(COMPILE.s) -o $@ $(COM_DIR)/amd64/$(@F:.o=.s)
103 $(POST_PROCESS_O)
104 @($(LHEAD) $(LINT.s) $(COM_DIR)/amd64/$(@F:.ln=.s) $(LTAIL))
```

new/usr/src/uts/intel/arcfour/Makefile

1

```
*****
2274 Thu Apr 25 02:20:50 2019
new/usr/src/uts/intel/arcfour/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 2014 Garrett D'Amore <garrett@damore.org>
22 #
23 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 #
26 # Copyright (c) 2019, Joyent, Inc.
27 # This makefile drives the production of the ARCFOUR KEF provider.
28 # Intel implementation architecture dependent
29 #
30 #
31 # Path to the base of the uts directory tree (usually /usr/src/uts).
32 #
33 # UTSBASE = ../../
34 # COM_DIR = $(COMMONBASE)/crypto/arcfour
35 #
36 # Define the module and object file sets.
37 #
38 # MODULE = arcfour
39 # LINTS = $(ARCFOURPROV_OBJS:%.o=$(LINTS_DIR)/%.ln)
40 # ARCFOURPROV_OBJS_32 =
41 # ARCFOURPROV_OBJS_64 = arcfour-x86_64.o
42 # ARCFOURPROV_OBJS += $(ARCFOURPROV_OBJS_$(CLASS))
43 # OBJECTS = $(ARCFOURPROV_OBJS:%=$(OBJSDIR)/%)
44 # ROOTMODULE = $(ROOT_CRYPTODIR)/$(MODULE)
45 #
46 #
47 # Include common rules.
48 #
49 include $(UTSBASE)/intel/Makefile.intel
```

new/usr/src/uts/intel/arcfour/Makefile

2

```
51 #
52 # Define targets
53 #
54 ALL_TARGET = $(BINARY)
55 LINT_TARGET = $(MODULE).lint
56 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)
57 #
58 #
59 # Overrides
60 #
61 CPPFLAGS += -I$(COM_DIR)
62 CFLAGS += -xO4 -xcrossfile
63 CLEANFILES += arcfour-x86_64.s
64 #
65 #
66 # Default build targets.
67 #
68 .KEEP_STATE:
69 #
70 def: $(DEF_DEPS)
71 #
72 all: $(ALL_DEPS)
73 #
74 clean: $(CLEAN_DEPS)
75 #
76 clobber: $(CLOBBER_DEPS)
77 #
78 lint: $(LINT_DEPS)
79 #
80 modlintlib: $(MODLINTLIB_DEPS)
81 #
82 clean.lint: $(CLEAN_LINT_DEPS)
83 #
84 install: $(INSTALL_DEPS)
85 #
86 #
87 # Include common targets.
88 #
89 include $(UTSBASE)/intel/Makefile.targ
90 #
91 $(OBJSDIR)/arcfour-x86_64.o: arcfour-x86_64.s
92 $(COMPILE.s) -o $@ ${@F:.o=.s}
93 $(POST_PROCESS_O)
94 #
95 $(OBJSDIR)/arcfour-x86_64.ln: arcfour-x86_64.s
96 @$(LHEAD) $(LINT.s) ${@F:.ln=.s} $(LTAIL)
97 #
98 arcfour-x86_64.s: $(COM_DIR)/amd64/arcfour-x86_64.pl
99 $(PERL) $? $@
```

```

*****
3440 Thu Apr 25 02:20:50 2019
new/usr/src/uts/intel/bignum/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 2009 Sun Microsystems, Inc. All rights reserved.
22 # Use is subject to license terms.
23 #
24 # Copyright (c) 2008, 2010, Oracle and/or its affiliates. All rights reserved.
25 #
26 # Copyright (c) 2019, Joyent, Inc.
27 # This makefile drives the production of the RSA KEF provider.
28 # intel implementation architecture dependent
29 #
30 # Path to the base of the uts directory tree (usually /usr/src/uts).
31 #
32 UTSBASE = ../../
33 BIGNUMDIR = $(COMMONBASE)/bignum
34 CRYPTODIR = $(COMMONBASE)/crypto
35 #
36 #
37 # Define the module and object file sets.
38 #
39 MODULE = bignum
40 OBJECTS = $(BIGNUM_OBJS:%=$(OBJS_DIR)/%)
41 LINTS = $(BIGNUM_OBJS:%.o=$(LINTS_DIR)/%.ln)
42 ROOTMODULE = $(ROOT_MISC_DIR)/$(MODULE)
43 #
44 BIGNUM_PSR_OBJS = bignum_amd64.o bignum_amd64_asm.o
44 BIGNUM_OBJS_32 = bignum_i386.o bignum_i386_asm.o
45 BIGNUM_OBJS_64 = bignum_amd64.o bignum_amd64_asm.o
46 BIGNUM_PSR_OBJS = $(BIGNUM_OBJS_$(CLASS))
47 #
48 # Include common rules.

```

```

48 #
49 include $(UTSBASE)/intel/Makefile.intel
50 #
51 #
52 # Define targets
53 #
54 ALL_TARGET = $(BINARY)
55 LINT_TARGET = $(MODULE).lint
56 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)
57 #
58 #
59 # Linkage dependencies
60 #
61 LDFLAGS += -dy -Nmisc/kcf
62 #
63 CPPFLAGS += -I$(BIGNUMDIR) -I$(CRYPTODIR)
64 #
65 #
66 # Default build targets.
67 #
68 .KEEP_STATE:
69 #
70 def: $(DEF_DEPS)
71 #
72 all: $(ALL_DEPS)
73 #
74 clean: $(CLEAN_DEPS)
75 #
76 clobber: $(CLOBBER_DEPS)
77 #
78 lint: $(LINT_DEPS)
79 #
80 modlintlib: $(MODLINTLIB_DEPS)
81 #
82 clean.lint: $(CLEAN_LINT_DEPS)
83 #
84 install: $(INSTALL_DEPS)
85 #
86 #
87 # Include common targets.
88 #
89 include $(UTSBASE)/intel/Makefile.targ
90 #
91 # Bignum configuration (BIGNUM_CFG):
92 # PSR_MUL:
93 # There is a processor-specific implementation bignum multiply functions
94 # HWCAP:
95 # There are multiple implementations of bignum functions, and the
96 # appropriate one must be chosen at run time, based on testing
97 # hardware capabilities.
98 #
99 # -DPSR_MUL:
100 # For AMD64, there is a processor-specific implementation of
101 # the bignum multiply functions, which takes advantage of the
102 # 64x64->128 bit multiply instruction.
103 #
104 # -UHWCAP:
105 # There is only one implementation, because the 128 bit multiply using
106 # general-purpose registers is faster than any MMX or SSE2 implementation.
107 #
108 BIGNUM_CFG = -DPSR_MUL
109 CFLAGS += -xO4 -xcrossfile
110 $(OBJS_DIR)/bignumimpl.o $(LINTS_DIR)/bignumimpl.ln := \
111 CPPFLAGS += $(BIGNUM_CFG)
112 $(OBJS_DIR)/bignum_amd64.o $(LINTS_DIR)/bignum_amd64.ln := \
113 CPPFLAGS += $(BIGNUM_CFG)

```

```
115 $(OBJS_DIR)/bignum_amd64.o: $(BIGNUMDIR)/amd64/bignum_amd64.c
116     $(COMPILE.c) -o $@ $(BIGNUM_CFG) $(BIGNUMDIR)/amd64/bignum_amd64.c
117     $(CTFCONVERT_O)

119 $(OBJS_DIR)/bignum_amd64_asm.o: $(BIGNUMDIR)/amd64/bignum_amd64_asm.s
120     $(COMPILE.s) -P -o $@ $(BIGNUM_CFG) \
121     $(BIGNUMDIR)/amd64/bignum_amd64_asm.s

123 $(LINTS_DIR)/bignum_amd64.ln: $(BIGNUMDIR)/amd64/bignum_amd64.c
124     @$(LHEAD) $(LINT.c) $(BIGNUMDIR)/amd64/bignum_amd64.c $(LTAIL))

126 $(LINTS_DIR)/bignum_amd64_asm.ln: $(BIGNUMDIR)/amd64/bignum_amd64_asm.s
127     @$(LHEAD) $(LINT.s) $(BIGNUMDIR)/amd64/bignum_amd64_asm.s $(LTAIL))
94 include Makefile.$(CLASS)
```

new/usr/src/uts/intel/elfexec/Makefile

1

```
*****
3405 Thu Apr 25 02:20:51 2019
new/usr/src/uts/intel/elfexec/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 # uts/intel/elfexec/Makefile
23 #
24 # Copyright 2006 Sun Microsystems, Inc. All rights reserved.
25 # Use is subject to license terms.
26 # Copyright 2019, Joyent, Inc.
26 # Copyright 2016 Joyent, Inc.
27 #
28 #
29 #
30 # This makefile drives the production of the elfexec exec kernel
31 # module.
32 #
33 # intel architecture dependent
34 #
35 #
36 #
37 # Path to the base of the uts directory tree (usually /usr/src/uts).
38 #
39 UTSBASE = ../..
40 #
41 #
42 # Define the module and object file sets.
43 #
44 MODULE = elfexec
45 ELFEXEC_OBJS_32 =
46 ELFEXEC_OBJS_64 = elf32.o elf32_notes.o old32_notes.o
47 ELFEXEC_OBJS += $(ELFEXEC_OBJS_$(CLASS))
48 OBJECTS = $(ELFEXEC_OBJS:%=$(OBJDIR)/%)
49 LINTS = $(ELFEXEC_OBJS:%.o=$(LINTSDIR)/%.ln)
50 ROOTMODULE = $(ROOT_EXEC_DIR)/$(MODULE)
51 #
52 #
```

new/usr/src/uts/intel/elfexec/Makefile

2

```
53 # Include common rules.
54 #
55 include $(UTSBASE)/intel/Makefile.intel
56 #
57 #
58 # Define targets
59 #
60 ALL_TARGET = $(BINARY)
61 LINT_TARGET = $(MODULE).lint
62 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)
63 #
64 INC_PATH += -I$(UTSBASE)/intel/ia32
65 #
66 #
67 # For now, disable these lint checks; maintainers should endeavor
68 # to investigate and remove these for maximum lint coverage.
69 # Please do not carry these forward to new Makefiles.
70 #
71 LINTTAGS += -erroff=E_BAD_PTR_CAST_ALIGN
72 LINTTAGS += -erroff=E_PTRDIFF_OVERFLOW
73 LINTTAGS += -erroff=E_ASSIGN_NARROW_CONV
74 #
75 CERRWARN += -_gcc=-Wno-parentheses
76 CERRWARN += -_gcc=-Wno-uninitialized
77 #
78 #
79 # Default build targets.
80 #
81 .KEEP_STATE:
82 #
83 def: $(DEF_DEPS)
84 #
85 all: $(ALL_DEPS)
86 #
87 clean: $(CLEAN_DEPS)
88 #
89 clobber: $(CLOBBER_DEPS)
90 #
91 lint: $(LINT_DEPS)
92 #
93 modlintlib: $(MODLINTLIB_DEPS)
94 #
95 clean.lint: $(CLEAN_LINT_DEPS)
96 #
97 install: $(INSTALL_DEPS)
98 #
99 #
100 # Include common targets.
101 #
102 include $(UTSBASE)/intel/Makefile.targ
103 #
104 .NO_PARALLEL: $(LINTS)
105 #
106 $(OBJDIR)/elf32.o: $(UTSBASE)/common/exec/elf/elf.c
107 $(COMPILE.c) -o $@ -D_ELF32_COMPAT $(UTSBASE)/common/exec/elf/elf.c
108 $(CTFCONVERT_O)
109 #
110 $(OBJDIR)/elf32_notes.o: $(UTSBASE)/common/exec/elf/elf_notes.c
111 $(COMPILE.c) -o $@ -D_ELF32_COMPAT $(UTSBASE)/common/exec/elf/elf_notes.c
112 $(CTFCONVERT_O)
113 #
114 $(OBJDIR)/old32_notes.o: $(UTSBASE)/common/exec/elf/old_notes.c
115 $(COMPILE.c) -o $@ -D_ELF32_COMPAT $(UTSBASE)/common/exec/elf/old_notes.c
116 $(CTFCONVERT_O)
117 #
118 $(LINTSDIR)/elf32.ln: $(UTSBASE)/common/exec/elf/elf.c
```

```
119      @$(LHEAD) $(LINT.c) -Celf32 -D_ELF32_COMPAT $(UTSBASE)/common/exec/elf/  
120      @$ (MV) $(@F) $@  
  
122 $(LINTS_DIR)/elf32_notes.ln: $(UTSBASE)/common/exec/elf/elf_notes.c  
123      @$(LHEAD) $(LINT.c) -Celf32_notes -D_ELF32_COMPAT $(UTSBASE)/common/exe  
124      @$ (MV) $(@F) $@  
  
126 $(LINTS_DIR)/old32_notes.ln: $(UTSBASE)/common/exec/elf/old_notes.c  
127      @$(LHEAD) $(LINT.c) -Cold32_notes -D_ELF32_COMPAT $(UTSBASE)/common/exe  
128      @$ (MV) $(@F) $@
```

new/usr/src/uts/intel/kcf/Makefile

1

```
*****
2389 Thu Apr 25 02:20:51 2019
new/usr/src/uts/intel/kcf/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 #
25 # Copyright (c) 2019, Joyent, Inc.
25 # Copyright (c) 2018, Joyent, Inc.

27 #
28 # Path to the base of the uts directory tree (usually /usr/src/uts).
29 #
30 UTSBASE = ../../
31 COM_DIR = $(COMMONBASE)/crypto

33 #
34 # Define the module and object file sets.
35 #
36 MODULE = kcf
37 LINTS = $(KCF_OBJS:%.o=$(LINTS_DIR)/%.ln)
38 KCF_OBJS_32 =
39 KCF_OBJS_64 = gcm_intel.o
40 KCF_OBJS += $(KCF_OBJS_$(CLASS))
41 OBJECTS = $(KCF_OBJS:%=$(OBJDIR)/%)
42 ROOTMODULE = $(ROOT_MISC_DIR)/$(MODULE)

44 #
45 # Include common rules.
46 #
47 include $(UTSBASE)/intel/Makefile.intel

49 #
50 # Define targets
51 #
52 ALL_TARGET = $(BINARY)
```

new/usr/src/uts/intel/kcf/Makefile

2

```
53 LINT_TARGET = $(MODULE).lint
54 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)

56 #
57 # Linkage dependencies
58 #

60 #
61 # lint pass one enforcement
62 #
63 CFLAGS += $(CCVERBOSE) -I$(COM_DIR)
64 AS_CPPFLAGS += -I../..$(PLATFORM)

66 LINTTAGS += -I$(COM_DIR)

68 CERRWARN += -_gcc=-Wno-switch
69 CERRWARN += -_gcc=-Wno-uninitialized
70 CERRWARN += -_gcc=-Wno-parentheses
71 CERRWARN += -_gcc=-Wno-unused-label

73 # needs work
74 SMOFF += all_func_returns,signed_integer_overflow_check

76 #
77 # Default build targets.
78 #
79 .KEEP_STATE:

81 def: $(DEF_DEPS)

83 all: $(ALL_DEPS)

85 clean: $(CLEAN_DEPS)

87 clobber: $(CLOBBER_DEPS)

89 lint: $(LINT_DEPS)

91 modlintlib: $(MODLINTLIB_DEPS)

93 clean.lint: $(CLEAN_LINT_DEPS)

95 install: $(INSTALL_DEPS)

97 #
98 # Include common targets.
99 #
100 include $(UTSBASE)/intel/Makefile.targ

102 $(OBJDIR)/%.o: $(COM_DIR)/modes/amd64/%.s
103 $(COMPILE.s) -o $@ $(COM_DIR)/modes/amd64/${@F:.o=.s}
104 $(POST_PROCESS_O)

105 $(OBJDIR)/%.ln: $(COM_DIR)/modes/amd64/%.s
106 @$(LHEAD) $(LINT.s) $(COM_DIR)/modes/amd64/${@F:.ln=.s} $(LTAIL)
```

new/usr/src/uts/intel/md5/Makefile

1

```
*****
2297 Thu Apr 25 02:20:51 2019
new/usr/src/uts/intel/md5/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 # uts/intel/md5/Makefile
23 #
24 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
25 # Use is subject to license terms.
26 #
27 # Copyright (c) 2019, Joyent, Inc.
27 # This makefile drives the production of the md5 crypto kernel module.
28 #
29 # intel architecture dependent
30 #
31 #
32 #
33 # Path to the base of the uts directory tree (usually /usr/src/uts).
34 #
35 #
36 #
37 # Define the module and object file sets.
38 #
39 MODULE = md5
40 MD5_OBJS_32 =
41 MD5_OBJS_64 = md5_amd64.o
42 MD5_OBJS += $(MD5_OBJS_$(CLASS))
43 OBJECTS = $(MD5_OBJS:%=$(OBJSDIR)/%)
44 LINTS = $(MD5_OBJS:%.o=$(LINTSDIR)/%.ln)
45 ROOTMODULE = $(ROOTCRYPTODIR)/$(MODULE)
46 ROOTLINK = $(ROOTMISCDIR)/$(MODULE)
47 #
48 #
49 # Include common rules.
50 #
```

new/usr/src/uts/intel/md5/Makefile

2

```
51 include $(UTSBASE)/intel/Makefile.intel
52 #
53 #
54 # Override defaults
55 #
56 CLEANFILES += md5_amd64.s
57 #
58 #
59 # Define targets
60 #
61 ALL_TARGET = $(BINARY)
62 LINT_TARGET = $(MODULE).lint
63 INSTALL_TARGET = $(BINARY) $(ROOTMODULE) $(ROOTLINK)
64 #
65 #
66 # md5 depends on the kcf framework
67 #
68 LDFLAGS += -dy -Nmisc/kcf
69 #
70 #
71 # Default build targets.
72 #
73 .KEEP_STATE:
74 #
75 def: $(DEF_DEPS)
76 #
77 all: $(ALL_DEPS)
78 #
79 clean: $(CLEAN_DEPS)
80 #
81 clobber: $(CLOBBER_DEPS)
82 #
83 lint: $(LINT_DEPS)
84 #
85 modlintlib: $(MODLINTLIB_DEPS)
86 #
87 clean.lint: $(CLEAN_LINT_DEPS)
88 #
89 install: $(INSTALL_DEPS)
90 #
91 $(ROOTLINK): $(ROOTMISC_DIR) $(ROOTMODULE)
92 -$(RM) $@; ln $(ROOTMODULE) $@
93 #
94 #
95 # Include common targets.
96 #
97 include $(UTSBASE)/intel/Makefile.targ
98 #
99 md5_amd64.s: $(COMDIR)/amd64/md5_amd64.pl
100 $(PERL) $? $@
101 #
102 $(OBJSDIR)/md5_amd64.o: md5_amd64.s
103 $(COMPILE.s) -o $@ md5_amd64.s
104 $(POST_PROCESS_O)
105 #
106 $(OBJSDIR)/md5_amd64.ln: md5_amd64.s
107 @($(LHEAD) $(LINT.c) md5_amd64.s $(LTAIL))
```


new/usr/src/uts/intel/shal/Makefile

1

```
*****
2329 Thu Apr 25 02:20:51 2019
new/usr/src/uts/intel/shal/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 # uts/intel/shal/Makefile
23 #
24 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
25 # Use is subject to license terms.
26 #
27 # Copyright (c) 2019, Joyent, Inc.
28 #
29 # This makefile drives the production of the shal crypto kernel module.
30 #
31 # intel architecture dependent
32 #
33 #
34 # Path to the base of the uts directory tree (usually /usr/src/uts).
35 #
36 UTSBASE = ../../
37 COMDIR = $(COMMONBASE)/crypto
38 #
39 # Define the module and object file sets.
40 #
41 MODULE = shal
42 SHAL_OBJS_32 =
43 SHAL_OBJS_64 = shal-x86_64.o
44 SHAL_OBJS += $(SHAL_OBJS_$(CLASS))
45 OBJECTS = $(SHAL_OBJS:%=$(OBJDIR)/%)
46 LINTS = $(SHAL_OBJS:%.o=$(LINTSDIR)/%.ln)
47 ROOTMODULE = $(ROOTCRYPTODIR)/$(MODULE)
48 ROOTLINK = $(ROOTMISCDIR)/$(MODULE)
49 #
50 #
51 # Include common rules.
```

new/usr/src/uts/intel/shal/Makefile

2

```
47 #
48 include $(UTSBASE)/intel/Makefile.intel
49 #
50 #
51 # Override defaults
52 #
53 CLEANFILES += shal-x86_64.s
54 #
55 #
56 # Linkage dependencies
57 #
58 LDFLAGS += -dy -Nmisc/kcf
59 #
60 CFLAGS += -I$(COMDIR)
61 LINTFLAGS += -I$(COMDIR)
62 #
63 #
64 # Define targets
65 #
66 #
67 ALL_TARGET = $(BINARY)
68 LINT_TARGET = $(MODULE).lint
69 INSTALL_TARGET = $(BINARY) $(ROOTMODULE) $(ROOTLINK)
70 #
71 #
72 # Default build targets.
73 #
74 .KEEP_STATE:
75 #
76 def: $(DEF_DEPS)
77 #
78 all: $(ALL_DEPS)
79 #
80 clean: $(CLEAN_DEPS)
81 #
82 clobber: $(CLOBBER_DEPS)
83 #
84 lint: $(LINT_DEPS)
85 #
86 modlintlib: $(MODLINTLIB_DEPS)
87 #
88 clean.lint: $(CLEAN_LINT_DEPS)
89 #
90 install: $(INSTALL_DEPS)
91 #
92 $(ROOTLINK): $(ROOTMISC_DIR) $(ROOTMODULE)
93 -$(RM) $@; ln $(ROOTMODULE) $@
94 #
95 #
96 # Include common targets.
97 #
98 include $(UTSBASE)/intel/Makefile.targ
99 #
100 $(OBJDIR)/shal-x86_64.o: shal-x86_64.s
101 $(COMPILE.s) -o $@ ${@F:.o=.s}
102 $(POST_PROCESS_O)
103 #
104 $(OBJDIR)/shal-x86_64.ln: shal-x86_64.s
105 @($(LHEAD) $(LINT.c) ${@F:.ln=.s} $(LTAIL))
106 shal-x86_64.s: $(COMDIR)/shal/amd64/shal-x86_64.pl
107 $(PERL) $? $@
```

```

*****
2479 Thu Apr 25 02:20:52 2019
new/usr/src/uts/intel/sha2/Makefile
10816 ctf_dwarf_convert_type() relies on un-initialized id
10817 ctfconvert -i option is mis-handled
10818 Improve ctfconvert error messages
10819 ctfconvert should handle empty dies
10820 ctfconvert -i never converts
10821 bad free in ctf_dwarf_init_die
10815 shouldn't build gcore.c as part of kmdb
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Jerry Jelinek <jerry.jelinek@joyent.com>
*****
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 # Copyright (c) 2019, Joyent, Inc.
26 # This makefile drives the production of the sha2 crypto kernel module.
27 # intel architecture dependent
28 #
29 #
30 # Path to the base of the uts directory tree (usually /usr/src/uts).
31 #
32 #
33 #
34 # Define the module and object file sets.
35 #
36 # MODULE = sha2
37 # SHA2_OBJS_32 =
38 # SHA2_OBJS_64 = sha512-x86_64.o sha256-x86_64.o
39 # SHA2_OBJS += $(SHA2_OBJS_$(CLASS))
40 # OBJECTS = $(SHA2_OBJS:%=$(OBJDIR)/%)
41 # LINTS = $(SHA2_OBJS:%.o=$(LINTSDIR)/%.ln)
42 # ROOTMODULE = $(ROOTCRYPTODIR)/$(MODULE)
43 # ROOTLINK = $(ROOTMISCDIR)/$(MODULE)
44 #
45 #
46 # Include common rules.
47 #
48 include $(UTSBASE)/intel/Makefile.intel

```

```

50 #
51 # Override defaults
52 #
53 CLEANFILES += sha512-x86_64.s sha256-x86_64.s

55 #
56 # Define targets
57 #
58 ALL_TARGET = $(BINARY)
59 LINT_TARGET = $(MODULE).lint
60 INSTALL_TARGET = $(BINARY) $(ROOTMODULE) $(ROOTLINK)

62 #
63 # Linkage dependencies
64 #
65 LDFLAGS += -dy -Nmisc/kcf

67 CFLAGS += -I$(COMDIR)
68 LINTFLAGS += -I$(COMDIR)

70 CERRWARN += -_gcc=-Wno-switch
71 CERRWARN += -_gcc=-Wno-uninitialized

73 #
74 # Default build targets.
75 #
76 .KEEP_STATE:

78 def: $(DEF_DEPS)

80 all: $(ALL_DEPS)

82 clean: $(CLEAN_DEPS)

84 clobber: $(CLOBBER_DEPS)

86 lint: $(LINT_DEPS)

88 modlintlib: $(MODLINTLIB_DEPS)

90 clean.lint: $(CLEAN_LINT_DEPS)

92 install: $(INSTALL_DEPS)

94 $(ROOTLINK): $(ROOTMISC_DIR) $(ROOTMODULE)
95 -$(RM) $@; ln $(ROOTMODULE) $@

97 #
98 # Include common targets.
99 #
100 include $(UTSBASE)/intel/Makefile.targ

102 $(OBJDIR)/%.o: %.s
103 $(COMPILE.s) -o $@ ${@F:.o=.s}
104 $(POST_PROCESS_O)

105 $(OBJDIR)/%.ln: %.s
106 @($(LHEAD) $(LINT.c) ${@F:.ln=.s} $(LTAIL))

108 sha512-x86_64.s: $(COMDIR)/sha2/amd64/sha512-x86_64.pl
109 $(PERL) $? $@

111 sha256-x86_64.s: $(COMDIR)/sha2/amd64/sha512-x86_64.pl
112 $(PERL) $? $@

```