```
**********************************************************
    9813 Thu Feb 13 22:45:13 2014
new/exception_lists/check_rtime
Add exceptions for objects in which the omnios build system includes bad
runpaths
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#

#
# Copyright (c) 2009, 2010, Oracle and/or its affiliates. All rights reserved.
#

# This file provides exceptions to the usual rules applied to ELF objects by
# check_rtime. All strings are Perl regular expressions that are compared to
# file paths. In addition to the standard Perl syntax, there is one extension:
#
#       MACH(dir)
#
# is expanded into a regular expression that matches the given
# directory, or a 64-bit subdirectory of the directory with the
# name of a 64-bit architecture. For example, MACH(lib) will match
# any of the following:
#
#       lib
#       lib/amd64
#       lib/sparcv9


# Directory hierarchies to skip completely
SKIP            ^usr/lib/libc/                  # optimized libc
SKIP            ^usr/lib/rcm/                   # 4426119
SKIP            ^usr/perl5/                     # alan’s taking care of these :-
SKIP            ^usr/src/                       # no objects in source code

# Individual files that we don’t examine
SKIP            ^boot/grub/bin/grub$
SKIP            ^usr/apache/libexec/mod_ipp.so$ # Apache loadable module
# USIII specific extns. cause ldd noise on USII bld. m/c
SKIP            ^usr/lib/fps/sun4u/UltraSPARC.*/fptest$
SKIP            ^usr/MACH(lib)/lddstub$         # lddstub has no dependencies
SKIP            ^usr/MACH(lib)/libssagent\.so\.1$       # 4328854
SKIP            ^usr/lib/MACH(iconv)/geniconvtbl.so$    # 4384329

# picl file exclusions (4385799)
SKIP            ^usr/platform/.*/libpsvcplugin_psr\.so\.1
SKIP            ^usr/platform/.*/libpsvcpolicy_psr\.so\.1
SKIP            ^usr/platform/.*/libpsvcpolicy\.so\.1
SKIP            ^usr/lib/sysevent/modules/picl_slm.so$

# Objects that are allowed to have executable data segments
EXEC_DATA       ^MACH(lib)/ld\.so\.1$
EXEC_DATA       ^lib/libc\.so\.1$       # 6524709, 32-bit, needed for x86 only
EXEC_DATA       ^lib/amd64/libumem\.so\.1$ # ptcumem
EXEC_DATA       ^lib/libumem\.so\.1$    # ptcumem
EXEC_DATA       ^opt/SUNWdtrt/tst/.*/ustack/tst\.helper\.exe$
EXEC_DATA       ^platform/.*/MACH(kernel)/unix$
EXEC_DATA       ^platform/.*/multiboot$
EXEC_DATA       ^MACH(lib)/libumem_trampoline\.so\.1$

# Objects that are allowed to have an executable stack
EXEC_STACK      ^platform/.*/MACH(kernel)/unix$
EXEC_STACK      ^platform/.*/multiboot$

# Objects for which we allow relocations to the text segment
TEXTREL         ^platform/.*/MACH(kernel)/unix$

# Directories and files that are allowed to have no direct bound symbols
NODIRECT        ^platform/.*/MACH(kernel)/unix$
NODIRECT        ^usr/ucb
NODIRECT        ^usr/4lib/sbcp$

# Identify any files that should be skipped when building a crle(1)
# configuration file.  As the hwcap libraries can be loop-back mounted onto
# libc, these can confuse crle(1) because of their identical dev/inode.
NOCRLEALT       ^usr/lib/libc/libc_hwcap[1-3].so.1$

# Files that should contain debugging information.
STAB    ^platform/.*/MACH(kernel)/unix$

# Files that are allowed undefined references
UNDEF_REF       ^usr/lib/libnisdb\.so\.2$
UNDEF_REF       ^usr/snadm/lib/libsvm\.so\.1$

# Objects allowed to have unused dependencies
UNUSED_DEPS     ^usr/lib/picl/plugins/          # require devtree dependencies
UNUSED_DEPS     ^usr/lib/libp   # profile libc makes libm an unused dep of libc

# libm.so.2 dependency
UNUSED_OBJ      unused object=.*MACH(libm)/libm_hwcap1\.so\.2

# Unused runpaths due to dlopen() use
UNUSED_RPATH    /usr/lib/fs/autofs.*\ from\ .automountd
UNUSED_RPATH    /etc/ppp/plugins.*\ from\ .*pppd
UNUSED_RPATH    /usr/lib/inet/ppp.*\ from\ .*pppd
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libipsecutil\.so\.1
UNUSED_RPATH    /usr/platform/.*rsmlib.*\ from\ .*librsm\.so\.2
UNUSED_RPATH    \$ORIGIN.*\ from\ .*fcode.so
UNUSED_RPATH    /opt/VRTSvxvm/lib.*\ from\ .*libdiskmgt\.so\.1

# Unused runpaths in picl code
UNUSED_RPATH    /usr/platform/.*\ from\ .*/usr/platform
UNUSED_RPATH    /usr/lib/picl/.*\ from\ .*/usr/platform
UNUSED_RPATH    /usr/platform/.*\ from\ .*/usr/lib/picl

# Unused runpaths in non-OSNET objects we can’t change
UNUSED_RPATH    /usr/lib/mps.*\ from\ .*libnss3\.so
UNUSED_RPATH    /usr/lib/mps.*\ from\ .*libnssutil3\.so
UNUSED_RPATH    /usr/lib/mps.*\ from\ .*libsmime3\.so
UNUSED_RPATH    /usr/lib/mps.*\ from\ .*libssl3\.so
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libdbus-1\.so\.3
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libdbus-glib-1\.so\.2
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libglib-2\.0\.so\.0
UNUSED_RPATH    /usr/X11/lib.*\ from\ .*libglib-2\.0\.so\.0
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libgobject-2\.0\.so\.0
UNUSED_RPATH    /usr/X11/lib.*\ from\ .*libgobject-2\.0\.so\.0
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libgthread-2\.0\.so\.0
UNUSED_RPATH    /usr/X11/lib.*\ from\ .*libgthread-2\.0\.so\.0
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libcrypto\.so\.0\.9\.8
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libnetsnmp\.so\.15
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libgcc_s\.so\.1
UNUSED_RPATH    /usr/ccs/lib.*\ from\ .*libgcc_s\.so\.1
UNUSED_RPATH    /usr/lib.*\ from\ .*libgcc_s\.so\.1
UNUSED_RPATH    /usr/postgres/8.3/lib.*\ from\ .*libpq\.so\.5
UNUSED_RPATH    /usr/sfw/lib.*\ from\ .*libpq\.so\.5
UNUSED_RPATH    /usr/lib.*\ from\ .*/usr/lib/mps
UNUSED_RPATH    /usr/ccs/lib.*\ from\ .*/usr/lib/mps
UNUSED_RPATH    /usr/gnu/lib.*\ from\ .*/usr/lib/libpython2\.6
UNUSED_RPATH    /usr/snadm/lib.*\ from\ .*/usr/snadm/lib/libspmicommon\.so\.1
#endif /* ! codereview */

# Unused runpaths for reasons not captured above
UNUSED_RPATH    /usr/lib/smbsrv.*\ from\ .*libsmb\.so\.1        # future needs
UNUSED_RPATH    /usr.*\ from\ .*tst\.gcc\.exe                   # gcc built


# Unreferenced objects of non-OSnet objects we can’t change
UNREF_OBJ       /lib.*\ of\ .*libcimapi\.so
UNREF_OBJ       /lib.*\ of\ .*libdbus-1\.so\.3
UNREF_OBJ       /lib.*\ of\ .*libdbus-glib-1\.so\.2
UNREF_OBJ       /lib.*\ of\ .*libgio-2.0\.so\.0
UNREF_OBJ       /lib.*\ of\ .*libglib-2.0\.so\.0
UNREF_OBJ       /lib.*\ of\ .*libgobject-2.0\.so\.0
UNREF_OBJ       /lib.*\ of\ .*libgthread-2\.0\.so\.0
UNREF_OBJ       /lib.*\ of\ .*libjvm\.so
UNREF_OBJ       /lib.*\ of\ .*libnetsnmp\.so\.30
UNREF_OBJ       /lib.*\ of\ .*libnetsnmpagent\.so\.30
UNREF_OBJ       /lib.*\ of\ .*libnetsnmpmibs\.so\.30
UNREF_OBJ       /lib.*\ of\ .*libnetsnmphelpers\.so\.30
UNREF_OBJ       /lib.*\ of\ .*libnspr4\.so
UNREF_OBJ       /lib.*\ of\ .*libpq\.so\.5
UNREF_OBJ       /lib.*\ of\ .*libsoftokn3\.so
UNREF_OBJ       /lib.*\ of\ .*libspmicommon\.so\.1
UNREF_OBJ       /lib.*\ of\ .*libspmocommon\.so\.1
UNREF_OBJ       /lib.*\ of\ .*libssl3\.so
UNREF_OBJ       /lib.*\ of\ .*libtspi\.so\.1
UNREF_OBJ       /lib.*\ of\ .*libxml2\.so\.2
UNREF_OBJ       /lib.*\ of\ .*libxslt\.so\.1
UNREF_OBJ       /lib.*\ of\ .*libpq\.so\.4
UNREF_OBJ       /lib.*\ of\ .*libpython2\.4\.so\.1\.0
UNREF_OBJ       /lib.*\ of\ .*libpython2\.6\.so\.1\.0
UNREF_OBJ       /libgcc_s.*\ of\ .*libstdc\+\+\.so\.6

# Unreferenced object of objects we can’t change for other reasons
UNREF_OBJ       /libmapmalloc\.so\.1;\ unused\ dependency\ of   # interposer
UNREF_OBJ       /libstdc\+\+\.so\.6;\ unused\ dependency\ of    # gcc build
UNREF_OBJ       /libm\.so\.2.*\ of\ .*libstdc\+\+\.so\.6        # gcc build
UNREF_OBJ       /libgcc_s\.so\.1.*\ of\ .*libstdc\+\+\.so\.6    # omnios gcc mix
UNREF_OBJ       /lib.*\ of\ .*/lib/picl/plugins/                # picl
UNREF_OBJ       /lib.*\ of\ .*kcfd                              # interposer
UNREF_OBJ       /libpkcs11\.so\.1; .*\ of\ .*libkmf\.so\.1      # interposed
# Referenced by the Studio build, not the GCC build.  GCC eliminates the unused
# statics which have the dependence.
UNREF_OBJ       /libc\.so\.1.*\ of\ .*kldap\.so\.1


# Objects that used to contain system functionalty that has since
# migrated to libc. We preserve these libraries as pure filters for
# backward compatability but nothing needs to link to them.
OLDDEP          libaio\.so\.1                   # onnv build 44
```

```
OLDDEP          libdl\.so\.1                    # on10 build 49
OLDDEP          libdoor\.so\.1                  # onnv build 12
OLDDEP          libintl\.so\.1                  # on297 build 7
OLDDEP          libpthread\.so\.1               # on10 build 53
OLDDEP          librt\.so\.1                    # onnv build 44
OLDDEP          libsched\.so\.1                 # on10 build 36
OLDDEP          libthread\.so\.1                # on10 build 53
OLDDEP          libw\.so\.1                     # on297 build 7

# Files for which we skip checking of duplicate addresses in the
# symbol sort sections. Such exceptions should be rare --- most code will
# not have duplicate addresses, since it takes assember or a "#pragma weak"
# to do such aliasing in C. C++ is different: The compiler generates aliases
# for implementation reasons, and the mangled names used to encode argument
# and return value types are difficult to handle well in mapfiles.
# Furthermore, the Sun compiler and gcc use different and incompatible
# name mangling conventions. Since ON must be buildable by either, we
# would have to maintain two sets of mapfiles for each such object.
# C++ use is rare in ON, so this is not worth pursuing.
#
NOSYMSORT       opt/SUNWdtrt/tst/common/pid/tst.weak2.exe       # DTrace test
NOSYMSORT       lib/amd64/libnsl\.so\.1                         # C++
NOSYMSORT       lib/sparcv9/libnsl\.so\.1                       # C++
NOSYMSORT       lib/sparcv9/libfru\.so\.1                       # C++
NOSYMSORT       usr/lib/lms                                     # C++
NOSYMSORT       ld\.so\.1                                       # libc_pic.a use
NOSYMSORT       lib/libsun_fc\.so\.1                            # C++
NOSYMSORT       lib/amd64/libsun_fc\.so\.1                      # C++
NOSYMSORT       lib/sparcv9/libsun_fc\.so\.1                    # C++
NOSYMSORT       usr/lib/amd64/libfru\.so\.1                     # C++


# The libprtdiag_psr.so.1 objects built under usr/src/lib/libprtdiag_psr
# are a family, all built using the same makefile, targeted at different
# sparc hardware variants. There are a small number of cases where this
# one size fits all approach causes an object to be linked against an
# unneeded library.
UNREF_OBJ       lib/(libdevinfo|libcfgadm)\.so\.1; .*\ of\ .*SUNW,Netra-CP2300/l
```

```
************************************************************
    2046 Thu Feb 13 22:45:13 2014
new/usr/src/cmd/boot/bootadm/Makefile
don't pass in lint libraries which no longer exist
gag lint errors that stem from lint libraries that the omnios build system does
************************************************************
```

```
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# Copyright 2011 Nexenta Systems, Inc. All rights reserved.
#

PROG= bootadm

SBINLINKS= $(PROG)

OBJS= bootadm.o bootadm_upgrade.o bootadm_hyper.o
SRCS = $(OBJS:.o=.c)

include ../Makefile.com


.KEEP_STATE:

LDLIBS_i386= -lfdisk
LDLIBS +=       -lnvpair -lgen -ladm -lefi -lscf -lbe -lzfs $(LDLIBS_$(MACH))

# There's no lint library for zlib, so only include this when building
$(PROG) :=      LDLIBS += -lz
LDLIBS +=       -lnvpair -lgen -ladm -lefi -lscf -lz -lbe -lzfs $(LDLIBS_$(MACH)

# Writing into string literals is incorrect.  We need to match gcc's
# behavior, which causes us to take SIGSEGV on such a write.
CFLAGS +=       $(XSTRCONST)
CPPFLAGS +=     -D_FILE_OFFSET_BITS=64
CPPFLAGS +=     -I../../../uts/common -I../../../common

CERRWARN +=     -_gcc=-Wno-parentheses
CERRWARN +=     -_gcc=-Wno-uninitialized
CERRWARN +=     -_gcc=-Wno-unused-label

LINTFLAGS += -erroff=E_INCONS_ARG_DECL2
LINTFLAGS += -erroff=E_INCONS_VAL_TYPE_DECL2
LINTFLAGS += -erroff=E_NAME_DEF_NOT_USED2
LINTFLAGS += -erroff=E_NAME_USED_NOT_DEF2
```

```
#endif /* ! codereview */

all: $(PROG)

$(PROG): $(OBJS) bootadm.h
        $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
        $(POST_PROCESS)

install: all $(ROOTSBINPROG) .WAIT $(ROOTUSRSBINLINKS)

clean:
        -$(RM) $(OBJS)

_msg:

lint:   lint_SRCS

include ../Makefile.targ
```

```
**********************************************************
    1527 Thu Feb 13 22:45:13 2014
new/usr/src/cmd/boot/fiocompress/Makefile
gag lint errors that stem from lint libraries that the omnios build system does
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
# Copyright 2007 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# ident "%Z%%M% %I%     %E% SMI"
#

PROG= fiocompress

SBINLINKS= $(PROG)

OBJS= fiocompress.o
SRCS = $(OBJS:.o=.c)

include ../Makefile.com

.KEEP_STATE:

LDLIBS +=       -lz

CFLAGS +=       -I../../../uts/common
LINTFLAGS +=    -I../../../uts/common

# definitions for lint
# until libz is compiled against ON header files (uid/gid)
LINTFLAGS += -erroff=E_INCONS_ARG_DECL2
LINTFLAGS += -erroff=E_INCONS_VAL_TYPE_DECL2
LINTFLAGS += -erroff=E_NAME_USED_NOT_DEF2
#endif /* ! codereview */

all: $(PROG)

$(PROG): $(OBJS)
        $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
        $(POST_PROCESS)

install: all $(ROOTSBINPROG) .WAIT $(ROOTUSRSBINLINKS)

clean:
        -$(RM) $(OBJS)
```

```
_msg:

lint:   lint_SRCS

include ../Makefile.targ
```

```
**********************************************************
    1566 Thu Feb 13 22:45:14 2014
new/usr/src/cmd/cmd-crypto/tpmadm/Makefile
don’t pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

PROG =  tpmadm

OBJS =  main.o \
        admin_cmds.o

include ../../Makefile.cmd

SRCS =  $(OBJS:%.o=%.c)
POFILES = $(OBJS:%.o=%.po)
POFILE = $(PROG)_msg.po
MSGFILES=$(SRCS:%.c=%.i)

CPPFLAGS += -xCC
LINTFLAGS += -XCC  -L$(ADJUNCT_PROTO)/usr/lib
LDLIBS += -luuid

# libtspi has no lint library, so we can only include this while building
$(PROG) := LDLIBS += -ltspi
LDLIBS += -ltspi -luuid

CLOBBERFILES = C.ln $(POFILE) $(POFILES)

.KEEP_STATE:

all : $(PROG)

$(PROG) : $(OBJS)
        $(LINK.c) -o $@ $(OBJS) $(DYNFLAGS) $(LDLIBS)
        $(POST_PROCESS)

$(POFILE) : $(POFILES)
        $(RM) $@; $(CAT) $(POFILES) > $@

install : all $(ROOTPROG)

clean :
        $(RM) $(OBJS)
```

```
lint : lint_SRCS

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

```
**********************************************************
    1265 Thu Feb 13 22:45:14 2014
new/usr/src/cmd/cmd-inet/usr.lib/wanboot/p12split/Makefile
don't pass in lint libraries which no longer exist
gag lint errors that stem from lint libraries that the omnios build system does
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

include ../Makefile.com

PROG=           p12split
LDLIBS +=       -lwanboot -linetutil -lwanbootutil
LDLIBS +=       -lwanboot -linetutil -lwanbootutil -lcrypto
CPPFLAGS +=     -I$(CMNCRYPTDIR)

# libcrypto has no lint library, so we can only include this while building
$(PROG) := LDLIBS += -lcrypto

LINTFLAGS +=    -erroff=E_NAME_USED_NOT_DEF2

#endif /* ! codereview */
all:            $(PROG)

install:        all $(ROOTCMD)

clean:

lint:           lint_PROG

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

****

15474 Thu Feb 13 22:45:15 2014

new/usr/src/cmd/cmd-inet/usr.lib/wanboot/p12split/p12split.c

remove lint supression made unnecessary by the openssl upgrade

****

```
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License, Version 1.0 only
 * (the "License").  You may not use this file except in compliance
 * with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright 2002-2003 Sun Microsystems, Inc.  All rights reserved.
 * Use is subject to license terms.
 */

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

#include <stdio.h>
#include <libintl.h>
#include <locale.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/wanboot_impl.h>
#include <unistd.h>
#include <string.h>
#include <libinetutil.h>
#include <wanbootutil.h>

#include <openssl/crypto.h>
#include <openssl/buffer.h>
#include <openssl/bio.h>
#include <openssl/err.h>
#include <openssl/x509.h>
#include <openssl/x509v3.h>
#include <openssl/pkcs12.h>
#include <openssl/evp.h>
#include <p12aux.h>

static boolean_t verbose = B_FALSE;     /* When nonzero, do in verbose mode */

/* The following match/cert values require PKCS12 */
static int  matchty;            /* Type of matching do to on input */
static char *k_matchval;        /* localkeyid value to match */
static uint_t k_len;            /* length of k_matchval */

#define IO_KEYFILE      1       /* Have a separate key file or data */
#define IO_CERTFILE     2       /* Have a separate cert file or data */
#define IO_TRUSTFILE    4       /* Have a separate trustanchor file */

static char *input = NULL;      /* Consolidated input file */
```

```
static char *key_out = NULL;    /* Key file to be output */
static char *cert_out = NULL;   /* Cert file to be output */
static char *trust_out = NULL;  /* Trust anchor file to be output */
static uint_t outfiles;         /* What files are there for output */
static char *progname;

/* Returns from time_check */
typedef enum {
        CHK_TIME_OK = 0,                /* Cert in effect and not expired */
        CHK_TIME_BEFORE_BAD,            /* not_before field is invalid */
        CHK_TIME_AFTER_BAD,             /* not_after field is invalid */
        CHK_TIME_IS_BEFORE,             /* Cert not yet in force */
        CHK_TIME_HAS_EXPIRED            /* Cert has expired */
} time_errs_t;
______unchanged_portion_omitted_

static int
do_certs(void)
{
        char *bufp;
        STACK_OF(X509) *ta_in = NULL;
        EVP_PKEY *pkey_in = NULL;
        X509 *xcert_in = NULL;

        sunw_crypto_init();

        if (read_files(&ta_in, &xcert_in, &pkey_in) < 0)
                return (-1);

        if (verbose) {
                if (xcert_in != NULL) {
                        (void) printf(gettext("\nMain cert:\n"));

                        /*
                         * sunw_subject_attrs() returns a pointer to
                         * memory allocated on our behalf. The same
                         * behavior is exhibited by sunw_issuer_attrs().
                         */
                        bufp = sunw_subject_attrs(xcert_in, NULL, 0);
                        if (bufp != NULL) {
                                (void) printf(gettext("  Subject: %s\n"),
                                    bufp);
                                OPENSSL_free(bufp);
                        }

                        bufp = sunw_issuer_attrs(xcert_in, NULL, 0);
                        if (bufp != NULL) {
                                (void) printf(gettext("  Issuer: %s\n"), bufp);
                                OPENSSL_free(bufp);
                        }

                        (void) sunw_print_times(stdout, PRNT_BOTH, NULL,
                            xcert_in);
                }

                if (ta_in != NULL) {
                        X509 *x;
                        int i;

                        for (i = 0; i < sk_X509_num(ta_in); i++) {
                                /* LINTED */
                                x = sk_X509_value(ta_in, i);
                                (void) printf(
                                    gettext("\nTrust Anchor cert %d:\n"), i);

                                /*
```

```
                                 * sunw_subject_attrs() returns a pointer to
                                 * memory allocated on our behalf. We get the
                                 * same behavior from sunw_issuer_attrs().
                                 */
                                bufp = sunw_subject_attrs(x, NULL, 0);
                                if (bufp != NULL) {
                                        (void) printf(
                                            gettext("  Subject: %s\n"), bufp);
                                        OPENSSL_free(bufp);
                                }

                                bufp = sunw_issuer_attrs(x, NULL, 0);
                                if (bufp != NULL) {
                                        (void) printf(
                                            gettext("  Issuer: %s\n"), bufp);
                                        OPENSSL_free(bufp);
                                }

                                (void) sunw_print_times(stdout, PRNT_BOTH,
                                    NULL, x);
                        }
                }
        }

        check_certs(ta_in, &xcert_in);
        if (xcert_in != NULL && pkey_in != NULL) {
                if (sunw_check_keys(xcert_in, pkey_in) == 0) {
                        wbku_printerr("warning: key and certificate do "
                            "not match\n");
                }
        }

        return (write_files(ta_in, xcert_in, pkey_in));
}
______unchanged_portion_omitted_

static void
check_certs(STACK_OF(X509) *ta_in, X509 **c_in)
{
        X509 *curr;
        time_errs_t ret;
        int i;
        int del_expired = (outfiles != 0);

        if (c_in != NULL && *c_in != NULL) {
                ret = time_check_print(*c_in);
                if ((ret != CHK_TIME_OK && ret != CHK_TIME_IS_BEFORE) &&
                    del_expired) {
                        (void) fprintf(stderr, gettext("  Removing cert\n"));
                        X509_free(*c_in);
                        *c_in = NULL;
                }
        }

        if (ta_in == NULL)
                return;

        for (i = 0; i < sk_X509_num(ta_in); ) {
                /* LINTED */
                curr = sk_X509_value(ta_in, i);
                ret = time_check_print(curr);
                if ((ret != CHK_TIME_OK && ret != CHK_TIME_IS_BEFORE) &&
                    del_expired) {
                        (void) fprintf(stderr, gettext("  Removing cert\n"));
                        /* LINTED */
                        curr = sk_X509_delete(ta_in, i);
```

```
                        X509_free(curr);
                        continue;
                }
                i++;
        }
}
______unchanged_portion_omitted_

static int
do_ofile(char *name, EVP_PKEY *pkey, X509 *cert, STACK_OF(X509) *ta)
{
        STACK_OF(EVP_PKEY) *klist = NULL;
        STACK_OF(X509)  *clist = NULL;
        PKCS12          *p12 = NULL;
        int             ret = 0;
        FILE            *fp;
        struct stat     sbuf;

        if (stat(name, &sbuf) == 0 && !S_ISREG(sbuf.st_mode)) {
                wbku_printerr("%s is not a regular file\n", name);
                return (-1);
        }

        if ((fp = fopen(name, "w")) == NULL) {
                wbku_printerr("cannot open output file %s", name);
                return (-1);
        }

        if ((clist = sk_X509_new_null()) == NULL ||
            (klist = sk_EVP_PKEY_new_null()) == NULL) {
                wbku_printerr("out of memory\n");
                ret = -1;
                goto cleanup;
        }

        if (cert != NULL && sk_X509_push(clist, cert) == 0) {
                wbku_printerr("out of memory\n");
                ret = -1;
                goto cleanup;
        }

        if (pkey != NULL && sk_EVP_PKEY_push(klist, pkey) == 0) {
                wbku_printerr("out of memory\n");
                ret = -1;
                goto cleanup;
        }

        p12 = sunw_PKCS12_create(WANBOOT_PASSPHRASE, klist, clist, ta);
        if (p12 == NULL) {
                wbku_printerr("cannot create %s: %s\n", name, cryptoerr());
                ret = -1;
                goto cleanup;
        }

        if (i2d_PKCS12_fp(fp, p12) == 0) {
                wbku_printerr("cannot write %s: %s\n", name, cryptoerr());
                ret = -1;
                goto cleanup;
        }

cleanup:
        (void) fclose(fp);
        if (p12 != NULL)
                PKCS12_free(p12);
        /*
         * Put the cert and pkey off of the stack so that they won't
```

```
	 * be freed two times.  (If they get left in the stack then
	 * they will be freed with the stack.)
	 */
	if (clist != NULL) {
		if (cert != NULL && sk_X509_num(clist) == 1) {
			/* LINTED */
			(void) sk_X509_delete(clist, 0);
		}
		sk_X509_pop_free(clist, X509_free);
	}
	if (klist != NULL) {
		if (pkey != NULL && sk_EVP_PKEY_num(klist) == 1) {
			/* LINTED */
			(void) sk_EVP_PKEY_delete(klist, 0);
		}
		sk_EVP_PKEY_pop_free(klist, sunw_evp_pkey_free);
	}

	return (ret);
}
```

______unchanged_portion_omitted_

```
**********************************************************
    1236 Thu Feb 13 22:45:15 2014
new/usr/src/cmd/cmd-inet/usr.lib/wanboot/wanboot-cgi/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

include ../Makefile.com

PROG =          wanboot-cgi
LDLIBS +=       -lgen -lnsl -lwanbootutil -lnvpair -lwanboot
LDLIBS +=       -lgen -lnsl -lwanbootutil -lnvpair -lwanboot -lcrypto
CPPFLAGS +=     -I$(CMNCRYPTDIR)

# libcrypto has no lint library, so we can only include this while building
$(PROG) := LDLIBS += -lcrypto

#endif /* ! codereview */
all:            $(PROG)

install:        all $(ROOTCMD)

clean:

lint:           lint_PROG

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

****
   45017 Thu Feb 13 22:45:16 2014
new/usr/src/cmd/cmd-inet/usr.lib/wanboot/wanboot-cgi/wanboot-cgi.c
remove lint supression made unnecessary by the openssl upgrade
****
______unchanged_portion_omitted_

```
/*
 * Add the certs found in the trustfile found in path (a trust store) to
 * the file found at bootfs_dir/truststore.  If necessary, create the
 * output file.
 */
static int
build_trustfile(const char *path, void *truststorepath)
{
        int             ret = WBCGI_FTW_CBERR;
        STACK_OF(X509)  *i_anchors = NULL;
        STACK_OF(X509)  *o_anchors = NULL;
        char            message[WBCGI_MAXBUF];
        PKCS12          *p12 = NULL;
        FILE            *rfp = NULL;
        FILE            *wfp = NULL;
        struct stat     i_st;
        struct stat     o_st;
        X509            *x = NULL;
        int             errtype = 0;
        int             wfd = -1;
        int             chars;
        int             i;

        if (!WBCGI_FILE_EXISTS(path, i_st)) {
                goto cleanup;
        }

        if (WBCGI_FILE_EXISTS((char *)truststorepath, o_st)) {
                /*
                 * If we are inadvertantly writing to the input file.
                 * return success.
                 * XXX Pete: how can this happen, and why success?
                 */
                if (i_st.st_ino == o_st.st_ino) {
                        ret = WBCGI_FTW_CBCONT;
                        goto cleanup;
                }
                if ((wfp = fopen((char *)truststorepath, "r+")) == NULL) {
                        goto cleanup;
                }
                /*
                 * Read what's already there, so that new information
                 * can be added.
                 */
                if ((p12 = d2i_PKCS12_fp(wfp, NULL)) == NULL) {
                        errtype = 1;
                        goto cleanup;
                }
                i = sunw_PKCS12_parse(p12, WANBOOT_PASSPHRASE, DO_NONE, NULL,
                    0, NULL, NULL, NULL, &o_anchors);
                if (i <= 0) {
                        errtype = 1;
                        goto cleanup;
                }

                PKCS12_free(p12);
                p12 = NULL;
        } else {
                if (errno != ENOENT) {
```

```
                        chars = snprintf(message, sizeof (message),
                            "(error accessing file %s, error %s)",
                            path, strerror(errno));
                        if (chars > 0 && chars < sizeof (message))
                                print_status(500, message);
                        else
                                print_status(500, NULL);
                        return (WBCGI_FTW_CBERR);
                }

                /*
                 * Note: We could copy the file to the new trustfile, but
                 * we can't verify the password that way.  Therefore, copy
                 * it by reading it.
                 */
                if ((wfd = open((char *)truststorepath,
                    O_CREAT|O_EXCL|O_RDWR, 0700)) < 0) {
                        goto cleanup;
                }
                if ((wfp = fdopen(wfd, "w+")) == NULL) {
                        goto cleanup;
                }
                o_anchors = sk_X509_new_null();
                if (o_anchors == NULL) {
                        goto cleanup;
                }
        }

        if ((rfp = fopen(path, "r")) == NULL) {
                goto cleanup;
        }
        if ((p12 = d2i_PKCS12_fp(rfp, NULL)) == NULL) {
                errtype = 1;
                goto cleanup;
        }
        i = sunw_PKCS12_parse(p12, WANBOOT_PASSPHRASE, DO_NONE, NULL, 0, NULL,
            NULL, NULL, &i_anchors);
        if (i <= 0) {
                errtype = 1;
                goto cleanup;
        }
        PKCS12_free(p12);
        p12 = NULL;

        /*
         * Merge the two stacks of pkcs12 certs.
         */
        for (i = 0; i < sk_X509_num(i_anchors); i++) {
                /* LINTED */
                x = sk_X509_delete(i_anchors, i);
                (void) sk_X509_push(o_anchors, x);
        }

        /*
         * Create the pkcs12 structure from the modified input stack and
         * then write out that structure.
         */
        p12 = sunw_PKCS12_create((const char *)WANBOOT_PASSPHRASE, NULL, NULL,
            o_anchors);
        if (p12 == NULL) {
                goto cleanup;
        }
        rewind(wfp);
        if (i2d_PKCS12_fp(wfp, p12) == 0) {
                goto cleanup;
        }
        ret = WBCGI_FTW_CBCONT;
cleanup:
        if (ret == WBCGI_FTW_CBERR) {
                if (errtype == 1) {
                        chars = snprintf(message, sizeof (message),
                            "(internal PKCS12 error while copying %s to %s)",
                            path, (char *)truststorepath);
                } else {
                        chars = snprintf(message, sizeof (message),
                            "(error copying %s to %s)",
                            path, (char *)truststorepath);
                }
                if (chars > 0 && chars <= sizeof (message)) {
                        print_status(500, message);
                } else {
                        print_status(500, NULL);
                }
        }
        if (rfp != NULL) {
                (void) fclose(rfp);
        }
        if (wfp != NULL) {
                /* Will also close wfd */
                (void) fclose(wfp);
        }
        if (p12 != NULL) {
                PKCS12_free(p12);
        }
        if (i_anchors != NULL) {
                sk_X509_pop_free(i_anchors, X509_free);
        }
        if (o_anchors != NULL) {
                sk_X509_pop_free(o_anchors, X509_free);
        }

        return (ret);
}
```

______***unchanged_portion_omitted_***

```
/*
 * Loop through the certificates in a file
 */
static int
get_hostnames(const char *path, void *nvl)
{
        int             ret = WBCGI_FTW_CBERR;
        STACK_OF(X509)  *certs = NULL;
        PKCS12          *p12 = NULL;
        char            message[WBCGI_MAXBUF];
        char            buf[WBCGI_MAXBUF + 1];
        FILE            *rfp = NULL;
        X509            *x = NULL;
        int             errtype = 0;
        int             chars;
        int             i;

        if ((rfp = fopen(path, "r")) == NULL) {
                goto cleanup;
        }

        if ((p12 = d2i_PKCS12_fp(rfp, NULL)) == NULL) {
                errtype = 1;
                goto cleanup;
        }
        i = sunw_PKCS12_parse(p12, WANBOOT_PASSPHRASE, DO_NONE, NULL, 0, NULL,
```

```
            NULL, NULL, &certs);
        if (i <= 0) {
                errtype = 1;
                goto cleanup;
        }

        PKCS12_free(p12);
        p12 = NULL;

        for (i = 0; i < sk_X509_num(certs); i++) {
                /* LINTED */
                x = sk_X509_value(certs, i);
                if (!one_name(sunw_issuer_attrs(x, buf, sizeof (buf) - 1),
                    nvl)) {
                        goto cleanup;
                }
        }

        ret = WBCGI_FTW_CBCONT;
cleanup:
        if (ret == WBCGI_FTW_CBERR) {
                if (errtype == 1) {
                        chars = snprintf(message, sizeof (message),
                            "(internal PKCS12 error reading %s)", path);
                } else {
                        chars = snprintf(message, sizeof (message),
                            "error reading %s", path);
                }
                if (chars > 0 && chars <= sizeof (message)) {
                        print_status(500, message);
                } else {
                        print_status(500, NULL);
                }
        }
        if (rfp != NULL) {
                (void) fclose(rfp);
        }
        if (p12 != NULL) {
                PKCS12_free(p12);
        }
        if (certs != NULL) {
                sk_X509_pop_free(certs, X509_free);
        }

        return (ret);
}
```

______***unchanged_portion_omitted_***

```
**********************************************************
    1517 Thu Feb 13 22:45:16 2014
new/usr/src/cmd/fm/modules/common/fabric-xlate/Makefile
don’t pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

MODULE = fabric-xlate
CLASS = common
SRCS = fabric-xlate.c fx_epkt.c fx_fabric.c fx_fire.c fx_subr.c

include ../../Makefile.plugin

CPPFLAGS += -I$(ADJUNCT_PROTO)/usr/include/libxml2 -I$(KMFDIR)/include -I.
CPPFLAGS += -I$(SRC)/uts/sun4v/io/px
INCDIRS = $(SRC)/uts/common
CFLAGS += -I$(INCDIRS)
LINTFLAGS += -I$(INCDIRS)

LDFLAGS += -L$(ROOT)/usr/lib/fm -R/usr/lib/fm
LDLIBS += -luutil -L$(ROOTLIB)/fm -ltopo

# libxml2 has no lint library, so we can only include this while building
$(PROG) := LDLIBS += -lxml2
LDLIBS += -luutil -lxml2 -L$(ROOTLIB)/fm -ltopo

CERRWARN += -_gcc=-Wno-parentheses
CERRWARN += -_gcc=-Wno-switch
CERRWARN += -_gcc=-Wno-uninitialized
```

```
**********************************************************
    1699 Thu Feb 13 22:45:17 2014
new/usr/src/cmd/isns/isnsadm/Makefile
don’t pass in lint libraries which no longer exist
gag lint errors that stem from lint libraries that the omnios build system does
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END

#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

PROG = isnsadm
SRCS = cmdparse.c isnsadm.c isnsadm_msg.c
OBJS = $(SRCS:%.c=%.o)

include ../../Makefile.cmd

LDLIBS += -lscf -lc
LDLIBS += -lxml2 -lscf -lc
CFLAGS += $(CCVERBOSE)
CPPFLAGS += -I. -I$(ADJUNCT_PROTO)/usr/include/libxml2 -I../isnsd
POFILE = isnsadm_all.po
POFILES = $(OBJS:%.o=%.po)

# libxml2 has no lint library, so we can only include this while building
$(PROG) := LDLIBS += -lxml2

#endif /* ! codereview */
LINTFLAGS += -xerroff=E_BAD_FORMAT_STR2
LINTFLAGS += -erroff=E_NAME_USED_NOT_DEF2
#endif /* ! codereview */

CERRWARN += -_gcc=-Wno-parentheses
CERRWARN += -_gcc=-Wno-uninitialized

.KEEP_STATE:

all: $(PROG)

clean:
        $(RM) $(OBJS) $(PROG)

lint:   lint_SRCS

$(PROG): $(OBJS)
        $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
```

```
        $(POST_PROCESS)

$(POFILE): $(POFILES)
        $(RM) $@
        $(CAT) $(POFILES) > $@

include ../../Makefile.targ

install: all .WAIT $(ROOTUSRSBINPROG)
```

```
**********************************************************
    2761 Thu Feb 13 22:45:17 2014
new/usr/src/cmd/isns/isnsd/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

#
# Copyright (c) 2012 by Delphix. All rights reserved.
#

PROG = isns

DSRC = isns_provider.d
DTRACE_HEADER = $(DSRC:%.d=%.h)

COBJS   = main.o qry.o admintf.o dsapi.o log.o sched.o dseng.o scn.o cache.o
COBJS   += dump.o msgq.o server.o config.o esi.o obj.o utils.o dd.o func.o
COBJS   += pdu.o door.o htable.o
OBJS    = $(COBJS) $(DSRC:%.d=%.o)
SRCS    = $(COBJS:%.o=%.c)
POFILE= isns.po
POFILES = $(COBJS:%.o=%.po)

include ../../Makefile.cmd

DATA_STORE      = xml
XML_DIR         = xml_def
ISNS_SVC        = $(XML_DIR)/isns_server.xml

MANIFEST        = isns_server.xml
ROOTMANIFESTDIR = $(ROOTSVCNETWORK)
$(ROOTSVCNETWORK)/isns_server.xml       := FILEMODE = 0444

CFLAGS          +=      $(CCVERBOSE)
CPPFLAGS        += -I./ -I$(ADJUNCT_PROTO)/usr/include/libxml2
CPPFLAGS        += -DTARGET_DATA_STORE=$(DATA_STORE)
CPPFLAGS        += -D_POSIX_PTHREAD_SEMANTICS
LINTFLAGS += -xerroff=E_BAD_PTR_CAST_ALIGN
LINTFLAGS += -xerroff=E_NAME_USED_NOT_DEF2
LINTFLAGS += -xerroff=E_CASE_FALLTHRU

CERRWARN += -_gcc=-Wno-char-subscripts
```

```
CERRWARN += -_gcc=-Wno-ignored-qualifiers
CERRWARN += -_gcc=-Wno-uninitialized
CERRWARN += -_gcc=-Wno-implicit-function-declaration

CLOBBERFILES += $(DTRACE_HEADER)

obj.o := CERRWARN += -erroff=E_CONST_OBJ_SHOULD_HAVE_INITIZR
obj.o := CERRWARN += -erroff=E_IMPLICIT_DECL_FUNC_RETURN_INT

LDLIBS          += -lsecdb -lsocket -lnsl -lscf

# libxml2 has no lint library, so we can only include this while building
$(PROG) := LDLIBS += -lxml2
LDLIBS          += -lsecdb -lsocket -lnsl -lscf -lxml2

.KEEP_STATE:

all:    $(PROG)

$(PROG): $(OBJS)
        $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
        $(POST_PROCESS)

%.h: %.d
        $(DTRACE) -h -s $< -o $@

%.o:    %.c $(DTRACE_HEADER)
        $(COMPILE.c) $<
        $(POST_PROCESS_O)

%.o:    %.d $(COBJS)
        $(COMPILE.d) -s $< $(COBJS)
        $(POST_PROCESS_O)

clean:
        $(RM) $(OBJS)

lint:   lint_SRCS

$(POFILE): $(POFILES)
        $(RM) $@
        $(CAT) $(POFILES) > $@

check:  $(CHKMANIFEST)

include ../../Makefile.targ

install: all .WAIT $(ROOTUSRSBINPROG) $(ROOTMANIFEST)
```

```
***********************************************************
   22988 Thu Feb 13 22:45:18 2014
new/usr/src/cmd/mdb/i86pc/modules/unix/unix.c
LINTED a bad cast align warning that seems to be oddly omnios specific.
I think the alignment of any pointer we receive from mdb_zalloc must be
sufficiently aligned for 'long' on both x86 platforms.
***********************************************************
______unchanged_portion_omitted_
```

```
/*
 * The x86 feature set is implemented as a bitmap array. That bitmap array is
 * stored across a number of uchars based on the BT_SIZEOFMAP(NUM_X86_FEATURES)
 * macro. We have the names for each of these features in unix's text segment
 * so we do not have to duplicate them and instead just look them up.
 */
/*ARGSUSED*/
static int
x86_featureset_cmd(uintptr_t addr, uint_t flags, int argc,
    const mdb_arg_t *argv)
{
        uchar_t *fset;
        GElf_Sym sym;
        uintptr_t nptr;
        char name[128];
        int ii;

        size_t sz = sizeof (uchar_t) * BT_SIZEOFMAP(NUM_X86_FEATURES);

        if (argc != 0)
                return (DCMD_USAGE);

        if (mdb_lookup_by_name("x86_feature_names", &sym) == -1) {
                mdb_warn("couldn't find x86_feature_names");
                return (DCMD_ERR);
        }

        fset = mdb_zalloc(sz, UM_NOSLEEP);
        if (fset == NULL) {
                mdb_warn("failed to allocate memory for x86_featureset");
                return (DCMD_ERR);
        }

        if (mdb_readvar(fset, "x86_featureset") != sz) {
                mdb_warn("failed to read x86_featureset");
                mdb_free(fset, sz);
                return (DCMD_ERR);
        }

        for (ii = 0; ii < NUM_X86_FEATURES; ii++) {
                /* LINTED */
#endif /* ! codereview */
                if (!BT_TEST((ulong_t *)fset, ii))
                        continue;

                if (mdb_vread(&nptr, sizeof (char *), sym.st_value +
                    sizeof (void *) * ii) != sizeof (char *)) {
                        mdb_warn("failed to read feature array %d", ii);
                        mdb_free(fset, sz);
                        return (DCMD_ERR);
                }

                if (mdb_readstr(name, sizeof (name), nptr) == -1) {
                        mdb_warn("failed to read feature %d", ii);
                        mdb_free(fset, sz);
                        return (DCMD_ERR);
                }
```

```
                mdb_printf("%s\n", name);
        }

        mdb_free(fset, sz);
        return (DCMD_OK);
}

static const mdb_dcmd_t dcmds[] = {
        { "gate_desc", ":", "dump a gate descriptor", gate_desc },
        { "idt", ":[-v]", "dump an IDT", idt },
        { "ttrace", "[-x]", "dump trap trace buffers", ttrace },
        { "vatopfn", ":[-a as]", "translate address to physical page",
            va2pfn_dcmd },
        { "report_maps", ":[-m]",
            "Given PFN, report mappings / page table usage",
            report_maps_dcmd, report_maps_help },
        { "htables", "", "Given hat_t *, lists all its htable_t * values",
            htables_dcmd, htables_help },
        { "ptable", ":[-m]", "Given PFN, dump contents of a page table",
            ptable_dcmd, ptable_help },
        { "pte", ":[-p XXXXX] [-l N]", "print human readable page table entry",
            pte_dcmd },
        { "pfntomfn", ":", "convert physical page to hypervisor machine page",
            pfntomfn_dcmd },
        { "mfntopfn", ":", "convert hypervisor machine page to physical page",
            mfntopfn_dcmd },
        { "memseg_list", ":", "show memseg list", memseg_list },
        { "scalehrtime", ":",
            "scale an unscaled high-res time", scalehrtime_cmd },
        { "x86_featureset", NULL, "dump the x86_featureset vector",
                x86_featureset_cmd },
        { NULL }
};

static const mdb_walker_t walkers[] = {
        { "ttrace", "walks trap trace buffers in reverse chronological order",
                ttrace_walk_init, ttrace_walk_step, ttrace_walk_fini },
        { "mutex_owner", "walks the owner of a mutex",
                mutex_owner_init, mutex_owner_step },
        { "memseg", "walk the memseg structures",
                memseg_walk_init, memseg_walk_step, memseg_walk_fini },
        { NULL }
};

static const mdb_modinfo_t modinfo = { MDB_API_VERSION, dcmds, walkers };

const mdb_modinfo_t *
_mdb_init(void)
{
        return (&modinfo);
}

void
_mdb_fini(void)
{
        free_mmu();
}
```

```
**********************************************************
   14127 Thu Feb 13 22:45:18 2014
new/usr/src/cmd/smbsrv/smbd/smbd_spool.c
Correctly merge "2991 Allow building without SMB printing support"
**********************************************************
/*
 * CDDL HEADER START
 *
 * The contents of this file are subject to the terms of the
 * Common Development and Distribution License (the "License").
 * You may not use this file except in compliance with the License.
 *
 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
 * or http://www.opensolaris.org/os/licensing.
 * See the License for the specific language governing permissions
 * and limitations under the License.
 *
 * When distributing Covered Code, include this CDDL HEADER in each
 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
 * If applicable, add the following below this CDDL HEADER, with the
 * fields enclosed by brackets "[]" replaced with your own identifying
 * information: Portions Copyright [yyyy] [name of copyright owner]
 *
 * CDDL HEADER END
 */
/*
 * Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
 * Copyright 2012 Nexenta Systems, Inc.  All rights reserved.
 */

/*
 * CUPS support for the SMB and SPOOLSS print services.
 */

#ifndef HAVE_CUPS
void smbd_load_printers(void) { return; }
void smbd_cups_init(void) { return; }
void smbd_cups_fini(void) { return; }
void smbd_spool_start(void) { return; }
void smbd_spool_stop(void) { return; }
#else

#include <sys/types.h>
#include <sys/stat.h>
#include <strings.h>
#include <syslog.h>
#include <signal.h>
#include <pthread.h>
#include <synch.h>
#include <dlfcn.h>
#include <errno.h>
#include <smbsrv/smb.h>
#include <smbsrv/smb_share.h>
#include "smbd.h"

#ifdef  HAVE_CUPS
#endif /* ! codereview */
#include <cups/cups.h>

#define SMB_SPOOL_WAIT                  2
#define SMBD_PJOBLEN                    256
#define SMBD_PRINTER                    "Postscript"
#define SMBD_FN_PREFIX                  "cifsprintjob-"
#define SMBD_CUPS_SPOOL_DIR             "//var//spool//cups"
#define SMBD_CUPS_DOCNAME               "generic_doc"
```

```
typedef struct smbd_printjob {
        pid_t           pj_pid;
        int             pj_sysjob;
        int             pj_fd;
        time_t          pj_start_time;
        int             pj_status;
        size_t          pj_size;
        int             pj_page_count;
        boolean_t       pj_isspooled;
        boolean_t       pj_jobnum;
        char            pj_filename[SMBD_PJOBLEN];
        char            pj_jobname[SMBD_PJOBLEN];
        char            pj_username[SMBD_PJOBLEN];
        char            pj_queuename[SMBD_PJOBLEN];
} smbd_printjob_t;

typedef struct smb_cups_ops {
        void            *cups_hdl;
        cups_lang_t     *(*cupsLangDefault)();
        const char      *(*cupsLangEncoding)(cups_lang_t *);
        void            (*cupsLangFree)(cups_lang_t *);
        ipp_status_t    (*cupsLastError)();
        int             (*cupsGetDests)(cups_dest_t **);
        void            (*cupsFreeDests)(int, cups_dest_t *);
        ipp_t           *(*cupsDoFileRequest)(http_t *, ipp_t *,
            const char *, const char *);
        ipp_t           *(*ippNew)();
        void            (*ippDelete)();
        char            *(*ippErrorString)();
        ipp_attribute_t *(*ippAddString)();
        void            (*httpClose)(http_t *);
        http_t          *(*httpConnect)(const char *, int);
} smb_cups_ops_t;

static uint32_t smbd_cups_jobnum = 1;
static smb_cups_ops_t smb_cups;
static mutex_t smbd_cups_mutex;

static void *smbd_spool_monitor(void *);
static smb_cups_ops_t *smbd_cups_ops(void);
static void smbd_print_share_comment(smb_share_t *, cups_dest_t *);
static void *smbd_share_printers(void *);
static void smbd_spool_copyfile(smb_inaddr_t *, char *, char *, char *);

extern smbd_t smbd;

/*
 * Start the spool thread.
 * Returns 0 on success, an error number if thread creation fails.
 */
void
smbd_spool_start(void)
{
        pthread_attr_t  attr;
        int             rc;

        if (!smb_config_getbool(SMB_CI_PRINT_ENABLE))
                return;

        (void) pthread_attr_init(&attr);
        (void) pthread_attr_setdetachstate(&attr, PTHREAD_CREATE_DETACHED);
        rc = pthread_create(&smbd.s_spool_tid, &attr, smbd_spool_monitor, NULL);
        (void) pthread_attr_destroy(&attr);

        if (rc != 0)
                smb_log(smbd.s_loghd, LOG_NOTICE,
                    "failed to start print monitor: %s", strerror(errno));
}

/*
 * A single pthread_kill should be sufficient but we include
 * a couple of retries to avoid implementation idiosyncrasies
 * around signal delivery.
 */
void
smbd_spool_stop(void)
{
        int     i;

        if (pthread_self() == smbd.s_spool_tid)
                return;

        for (i = 0; i < 3 && smbd.s_spool_tid != 0; ++i) {
                if (pthread_kill(smbd.s_spool_tid, SIGTERM) == ESRCH)
                        break;

                (void) sleep(1);
        }
}

/*
 * This thread blocks waiting for close print file in the kernel.
 * It then uses the data returned from the ioctl to copy the spool file
 * into the cups spooler.
 *
 * This mechanism is really only used by Windows Vista and Windows 7.
 * Other versions of Windows create a zero size file, which is removed
 * by smbd_spool_copyfile.
 */
/*ARGSUSED*/
static void *
smbd_spool_monitor(void *arg)
{
        uint32_t        spool_num;
        char            username[MAXNAMELEN];
        char            path[MAXPATHLEN];
        smb_inaddr_t    ipaddr;
        int             error_retry_cnt = 5;

        smbd_online_wait("smbd_spool_monitor");

        spoolss_register_copyfile(smbd_spool_copyfile);

        while (!smbd.s_shutting_down && (error_retry_cnt > 0)) {
                errno = 0;

                if (smb_kmod_get_spool_doc(&spool_num, username,
                    path, &ipaddr) == 0) {
                        smbd_spool_copyfile(&ipaddr,
                            username, path, SMBD_CUPS_DOCNAME);
                        error_retry_cnt = 5;
                } else {
                        if (errno == ECANCELED)
                                break;
                        if ((errno != EINTR) && (errno != EAGAIN))
                                error_retry_cnt--;
                        (void) sleep(SMB_SPOOL_WAIT);
                }
        }

        spoolss_register_copyfile(NULL);
        smbd.s_spool_tid = 0;
```

```
        return (NULL);
}

/*
 * All versions of windows use this function to spool files to a printer
 * via the cups interface
 */
static void
smbd_spool_copyfile(smb_inaddr_t *ipaddr, char *username, char *path,
    char *doc_name)
{
        smb_cups_ops_t  *cups;
        http_t          *http = NULL;           /* HTTP connection to server */
        ipp_t           *request = NULL;        /* IPP Request */
        ipp_t           *response = NULL;       /* IPP Response */
        cups_lang_t     *language = NULL;       /* Default language */
        char            uri[HTTP_MAX_URI];      /* printer-uri attribute */
        char            new_jobname[SMBD_PJOBLEN];
        smbd_printjob_t pjob;
        char            clientname[INET6_ADDRSTRLEN];
        struct stat     sbuf;
        int             rc = 1;

        if (stat(path, &sbuf)) {
                smb_log(smbd.s_loghd, LOG_INFO, "smbd_spool_copyfile: %s: %s",
                    path, strerror(errno));
                return;
        }

        /*
         * Remove zero size files and return; these were inadvertantly
         * created by XP or 2000.
         */
        if (sbuf.st_size == 0) {
                if (remove(path) != 0)
                        smb_log(smbd.s_loghd, LOG_INFO,
                            "smbd_spool_copyfile: cannot remove %s: %s",
                            path, strerror(errno));
                return;
        }

        if ((cups = smbd_cups_ops()) == NULL)
                return;

        if ((http = cups->httpConnect("localhost", 631)) == NULL) {
                smb_log(smbd.s_loghd, LOG_INFO,
                    "smbd_spool_copyfile: cupsd not running");
                return;
        }

        if ((request = cups->ippNew()) == NULL) {
                smb_log(smbd.s_loghd, LOG_INFO,
                    "smbd_spool_copyfile: ipp not running");
                return;
        }

        request->request.op.operation_id = IPP_PRINT_JOB;
        request->request.op.request_id = 1;
        language = cups->cupsLangDefault();

        cups->ippAddString(request, IPP_TAG_OPERATION, IPP_TAG_CHARSET,
            "attributes-charset", NULL, cups->cupsLangEncoding(language));

        cups->ippAddString(request, IPP_TAG_OPERATION, IPP_TAG_LANGUAGE,
            "attributes-natural-language", NULL, language->language);
        (void) snprintf(uri, sizeof (uri), "ipp://localhost/printers/%s",
            SMBD_PRINTER);
        pjob.pj_pid = pthread_self();
        pjob.pj_sysjob = 10;
        (void) strlcpy(pjob.pj_filename, path, SMBD_PJOBLEN);
        pjob.pj_start_time = time(NULL);
        pjob.pj_status = 2;
        pjob.pj_size = sbuf.st_blocks * 512;
        pjob.pj_page_count = 1;
        pjob.pj_isspooled = B_TRUE;
        pjob.pj_jobnum = smbd_cups_jobnum;

        (void) strlcpy(pjob.pj_jobname, doc_name, SMBD_PJOBLEN);
        (void) strlcpy(pjob.pj_username, username, SMBD_PJOBLEN);
        (void) strlcpy(pjob.pj_queuename, SMBD_CUPS_SPOOL_DIR, SMBD_PJOBLEN);

        cups->ippAddString(request, IPP_TAG_OPERATION, IPP_TAG_URI,
            "printer-uri", NULL, uri);

        cups->ippAddString(request, IPP_TAG_OPERATION, IPP_TAG_NAME,
            "requesting-user-name", NULL, pjob.pj_username);

        if (smb_inet_ntop(ipaddr, clientname,
            SMB_IPSTRLEN(ipaddr->a_family)) == NULL) {
                smb_log(smbd.s_loghd, LOG_INFO,
                    "smbd_spool_copyfile: %s: unknown client", clientname);
                goto out;
        }

        cups->ippAddString(request, IPP_TAG_OPERATION, IPP_TAG_NAME,
            "job-originating-host-name", NULL, clientname);

        (void) snprintf(new_jobname, SMBD_PJOBLEN, "%s%d",
            SMBD_FN_PREFIX, pjob.pj_jobnum);
        cups->ippAddString(request, IPP_TAG_OPERATION, IPP_TAG_NAME,
            "job-name", NULL, new_jobname);

        (void) snprintf(uri, sizeof (uri) - 1, "/printers/%s", SMBD_PRINTER);

        response = cups->cupsDoFileRequest(http, request, uri,
            pjob.pj_filename);
        if (response != NULL) {
                if (response->request.status.status_code >= IPP_OK_CONFLICT) {
                        smb_log(smbd.s_loghd, LOG_ERR,
                            "smbd_spool_copyfile: printer %s: %s",
                            SMBD_PRINTER,
                            cups->ippErrorString(cups->cupsLastError()));
                } else {
                        atomic_inc_32(&smbd_cups_jobnum);
                        rc = 0;
                }
        } else {
                smb_log(smbd.s_loghd, LOG_ERR,
                    "smbd_spool_copyfile: unable to print to %s",
                    cups->ippErrorString(cups->cupsLastError()));
        }

        if (rc == 0)
                (void) unlink(pjob.pj_filename);

out:
        if (response)
                cups->ippDelete(response);

        if (language)
                cups->cupsLangFree(language);
```

```

        if (http)
                cups->httpClose(http);
}

int
smbd_cups_init(void)
{
        (void) mutex_lock(&smbd_cups_mutex);

        if (smb_cups.cups_hdl != NULL) {
                (void) mutex_unlock(&smbd_cups_mutex);
                return (0);
        }

        if ((smb_cups.cups_hdl = dlopen("libcups.so.2", RTLD_NOW)) == NULL) {
                (void) mutex_unlock(&smbd_cups_mutex);
                smb_log(smbd.s_loghd, LOG_DEBUG,
                    "smbd_cups_init: cannot open libcups");
                return (ENOENT);
        }

        smb_cups.cupsLangDefault =
            (cups_lang_t *(*)())dlsym(smb_cups.cups_hdl, "cupsLangDefault");
        smb_cups.cupsLangEncoding = (const char *(*)(cups_lang_t *))
            dlsym(smb_cups.cups_hdl, "cupsLangEncoding");
        smb_cups.cupsDoFileRequest =
            (ipp_t *(*)(http_t *, ipp_t *, const char *, const char *))
            dlsym(smb_cups.cups_hdl, "cupsDoFileRequest");
        smb_cups.cupsLastError = (ipp_status_t (*)())
            dlsym(smb_cups.cups_hdl, "cupsLastError");
        smb_cups.cupsLangFree = (void (*)(cups_lang_t *))
            dlsym(smb_cups.cups_hdl, "cupsLangFree");
        smb_cups.cupsGetDests = (int (*)(cups_dest_t **))
            dlsym(smb_cups.cups_hdl, "cupsGetDests");
        smb_cups.cupsFreeDests = (void (*)(int, cups_dest_t *))
            dlsym(smb_cups.cups_hdl, "cupsFreeDests");

        smb_cups.httpClose = (void (*)(http_t *))
            dlsym(smb_cups.cups_hdl, "httpClose");
        smb_cups.httpConnect = (http_t *(*)(const char *, int))
            dlsym(smb_cups.cups_hdl, "httpConnect");

        smb_cups.ippNew = (ipp_t *(*)())dlsym(smb_cups.cups_hdl, "ippNew");
        smb_cups.ippDelete = (void (*)())dlsym(smb_cups.cups_hdl, "ippDelete");
        smb_cups.ippErrorString = (char *(*)())
            dlsym(smb_cups.cups_hdl, "ippErrorString");
        smb_cups.ippAddString = (ipp_attribute_t *(*)())
            dlsym(smb_cups.cups_hdl, "ippAddString");

        if (smb_cups.cupsLangDefault == NULL ||
            smb_cups.cupsLangEncoding == NULL ||
            smb_cups.cupsDoFileRequest == NULL ||
            smb_cups.cupsLastError == NULL ||
            smb_cups.cupsLangFree == NULL ||
            smb_cups.cupsGetDests == NULL ||
            smb_cups.cupsFreeDests == NULL ||
            smb_cups.ippNew == NULL ||
            smb_cups.httpClose == NULL ||
            smb_cups.httpConnect == NULL ||
            smb_cups.ippDelete == NULL ||
            smb_cups.ippErrorString == NULL ||
            smb_cups.ippAddString == NULL) {
                (void) dlclose(smb_cups.cups_hdl);
                smb_cups.cups_hdl = NULL;
                (void) mutex_unlock(&smbd_cups_mutex);
                smb_log(smbd.s_loghd, LOG_DEBUG,
                    "smbd_cups_init: cannot load libcups");
                return (ENOENT);
        }

        (void) mutex_unlock(&smbd_cups_mutex);
        return (0);
}

void
smbd_cups_fini(void)
{
        (void) mutex_lock(&smbd_cups_mutex);

        if (smb_cups.cups_hdl != NULL) {
                (void) dlclose(smb_cups.cups_hdl);
                smb_cups.cups_hdl = NULL;
        }

        (void) mutex_unlock(&smbd_cups_mutex);
}

static smb_cups_ops_t *
smbd_cups_ops(void)
{
        if (smb_cups.cups_hdl == NULL)
                return (NULL);

        return (&smb_cups);
}

void
smbd_load_printers(void)
{
        pthread_t       tid;
        pthread_attr_t  attr;
        int             rc;

        if (!smb_config_getbool(SMB_CI_PRINT_ENABLE))
                return;

        (void) pthread_attr_init(&attr);
        (void) pthread_attr_setdetachstate(&attr, PTHREAD_CREATE_DETACHED);
        rc = pthread_create(&tid, &attr, smbd_share_printers, &tid);
        (void) pthread_attr_destroy(&attr);

        if (rc != 0)
                smb_log(smbd.s_loghd, LOG_NOTICE,
                    "unable to load printer shares: %s", strerror(errno));
}

/*
 * All print shares use the path from print$.
 */
/*ARGSUSED*/
static void *
smbd_share_printers(void *arg)
{
        cups_dest_t     *dests;
        cups_dest_t     *dest;
        smb_cups_ops_t  *cups;
        smb_share_t     si;
        uint32_t        nerr;
        int             num_dests;
        int             i;
```

```
        if (!smb_config_getbool(SMB_CI_PRINT_ENABLE))
                return (NULL);

        if ((cups = smbd_cups_ops()) == NULL)
                return (NULL);

        if (smb_shr_get(SMB_SHARE_PRINT, &si) != NERR_Success) {
                smb_log(smbd.s_loghd, LOG_DEBUG,
                    "smbd_share_printers unable to load %s", SMB_SHARE_PRINT);
                return (NULL);
        }

        num_dests = cups->cupsGetDests(&dests);

        for (i = num_dests, dest = dests; i > 0; i--, dest++) {
                if (dest->instance != NULL)
                        continue;

                (void) strlcpy(si.shr_name, dest->name, MAXPATHLEN);
                smbd_print_share_comment(&si, dest);
                si.shr_type = STYPE_PRINTQ;

                nerr = smb_shr_add(&si);
                if (nerr == NERR_Success || nerr == NERR_DuplicateShare)
                        smb_log(smbd.s_loghd, LOG_DEBUG,
                            "shared printer: %s", si.shr_name);
                else
                        smb_log(smbd.s_loghd, LOG_DEBUG,
                            "smbd_share_printers: unable to add share %s: %u",
                            si.shr_name, nerr);
        }

        cups->cupsFreeDests(num_dests, dests);
        return (NULL);
}

static void
smbd_print_share_comment(smb_share_t *si, cups_dest_t *dest)
{
        cups_option_t   *options;
        char            *comment;
        char            *name;
        char            *value;
        int             i;

        comment = "Print Share";

        if ((options = dest->options) == NULL) {
                (void) strlcpy(si->shr_cmnt, comment, SMB_SHARE_CMNT_MAX);
                return;
        }

        for (i = 0; i < dest->num_options; ++i) {
                name = options[i].name;
                value = options[i].value;

                if (name == NULL || value == NULL ||
                    *name == '\0' || *value == '\0')
                        continue;

                if (strcasecmp(name, "printer-info") == 0) {
                        comment = value;
                        break;
                }
        }
        (void) strlcpy(si->shr_cmnt, comment, SMB_SHARE_CMNT_MAX);
}

#else   /* HAVE_CUPS */

/*
 * If not HAVE_CUPS, just provide a few "stubs".
 */

int
smbd_cups_init(void)
{
        return (ENOENT);
}

void
smbd_cups_fini(void)
{
}

void
smbd_load_printers(void)
{
}

void
smbd_spool_init(void)
{
}

void
smbd_spool_fini(void)
{
}

void
smbd_spool_start(void)
{
}

void
smbd_spool_stop(void)
{
}

#endif  /* HAVE_CUPS */
#endif
```

```
**********************************************************
    2440 Thu Feb 13 22:45:19 2014
new/usr/src/cmd/ssh/libssh/Makefile.com
don't pass in lint libraries which no longer exist
**********************************************************
```

```
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

LIBRARY =       libssh.a
VERS =          .1

OBJECTS =       \
                addrmatch.o \
                atomicio.o \
                authfd.o \
                authfile.o \
                bufaux.o \
                buffer.o \
                canohost.o \
                channels.o \
                cipher.o \
                cipher-ctr.o \
                compat.o \
                compress.o \
                crc32.o \
                deattack.o \
                dh.o \
                dispatch.o \
                engine.o \
                entropy.o \
                fatal.o \
                g11n.o \
                hostfile.o \
                key.o \
                kex.o \
                kexdh.o \
                kexdhc.o \
                kexdhs.o \
                kexgex.o \
                kexgexc.o \
                kexgexs.o \
                kexgssc.o \
                kexgsss.o \
                log.o \
                mac.o \
```

```
                match.o \
                misc.o \
                mpaux.o \
                msg.o \
                nchan.o \
                packet.o \
                progressmeter.o \
                proxy-io.o \
                radix.o \
                readconf.o \
                readpass.o \
                rsa.o \
                sftp-common.o \
                ssh-dss.o \
                ssh-gss.o \
                ssh-rsa.o \
                tildexpand.o \
                ttymodes.o \
                uidswap.o \
                uuencode.o \
                xlist.o \
                xmalloc.o

include $(SRC)/lib/Makefile.lib

BUILD.AR =       $(RM) $@ ; $(AR) $(ARFLAGS) $@ $(AROBJS)

SRCDIR =        ../common
SRCS =          $(OBJECTS:%.o=../common/%.c)

LIBS =          $(LIBRARY) $(LINTLIB)

# Define LDLIBS conditionally for lintcheck, rather than in general, since
# we're building an archive library which itself links to nothing, we just
# want lint to know about these libraries.
lintcheck  :=   LDLIBS += -lsocket -lnsl -lc
lintcheck  :=   LDLIBS += -lcrypto -lz -lsocket -lnsl -lc
$(LINTLIB) :=   SRCS = $(SRCDIR)/$(LINTSRC)

POFILE_DIR =    ../..

.KEEP_STATE:

all:            $(LIBS)

# lint requires the (not installed) lint library
lint:           $(LINTLIB) .WAIT lintcheck

include $(SRC)/lib/Makefile.targ

objs/%.o:       $(SRCDIR)/%.c
        $(COMPILE.c) -o $@ $<
        $(POST_PROCESS_O)

include ../../Makefile.ssh-common
include ../../Makefile.msg.targ
```

```
**********************************************************
    1595 Thu Feb 13 22:45:19 2014
new/usr/src/cmd/ssh/sftp-server/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# cmd/ssh/sftp-server/Makefile

PROG =          sftp-server

OBJS =          sftp-server.o sftp-server-main.o
SRCS =          $(OBJS:.o=.c)

include ../../Makefile.cmd
include ../Makefile.ssh-common

LDLIBS +=       $(SSH_COMMON_LDLIBS) -lsocket

# libcrypto has no lint library, so we can only use it when building
$(PROG) :=      LDLIBS += -lcrypto
LDLIBS +=       $(SSH_COMMON_LDLIBS) -lsocket -lcrypto

POFILE_DIR =    ..

.KEEP_STATE:

.PARALLEL:      $(OBJS)

all:            $(PROG)

$(PROG):        $(OBJS) ../libssh/$(MACH)/libssh.a ../libopenbsd-compat/$(MACH)/
        $(LINK.c) $(OBJS) -o $@ $(LDLIBS) $(DYNFLAGS)
        $(POST_PROCESS)

install:        all $(ROOTLIBSSHPROG) $(ROOTLIBSSH)

clean:
        $(RM) -f $(OBJS) $(PROG)

lint:           lint_SRCS

include ../Makefile.msg.targ
include ../../Makefile.targ
```

```
**********************************************************
    1579 Thu Feb 13 22:45:19 2014
new/usr/src/cmd/ssh/sftp/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# cmd/ssh/sftp/Makefile

PROG =          sftp

OBJS =          \
                sftp.o \
                sftp-client.o \
                sftp-glob.o

SRCS =          $(OBJS:.o=.c)

include ../../Makefile.cmd
include ../Makefile.ssh-common

LDLIBS +=       $(SSH_COMMON_LDLIBS) -lsocket -ltecla
# libcrypto has no lint library, so we can only use it when building
$(PROG) := LDLIBS += -lcrypto
LDLIBS +=       $(SSH_COMMON_LDLIBS) -lsocket -lcrypto -ltecla

POFILE_DIR =    ..

.KEEP_STATE:

.PARALLEL:      $(OBJS)

all: $(PROG)

$(PROG):        $(OBJS) ../libssh/$(MACH)/libssh.a ../libopenbsd-compat/$(MACH)/
        $(LINK.c) $(OBJS) -o $@ $(LDLIBS) $(DYNFLAGS)
        $(POST_PROCESS)

install:        all $(ROOTPROG)

clean:
        $(RM) -f $(OBJS) $(PROG)

lint:           lint_SRCS
```

```

include ../Makefile.msg.targ
include ../../Makefile.targ
```

```
**********************************************************
    1542 Thu Feb 13 22:45:20 2014
new/usr/src/cmd/ssh/ssh-add/Makefile
don’t pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2008 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# cmd/ssh/ssh-add/Makefile

PROG= ssh-add

OBJS    =        \
        ssh-add.o
SRCS    = $(OBJS:.o=.c)

include ../../Makefile.cmd
include ../Makefile.ssh-common

LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket

# libcrypto has no lint library, so we can only use it when building
$(PROG) := LDLIBS += -lcrypto
LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket -lcrypto

POFILE_DIR= ..

.KEEP_STATE:

.PARALLEL: $(OBJS)

all: $(PROG)

$(PROG): $(OBJS) ../libssh/$(MACH)/libssh.a ../libopenbsd-compat/$(MACH)/libopen
        $(LINK.c) $(OBJS) -o $@ $(LDLIBS) $(DYNFLAGS)
        $(POST_PROCESS)

install: all $(ROOTPROG)

clean:
        $(RM) -f $(OBJS) $(PROG)

lint:   lint_SRCS

include ../Makefile.msg.targ
include ../../Makefile.targ
```

```
**********************************************************
    1548 Thu Feb 13 22:45:20 2014
new/usr/src/cmd/ssh/ssh-agent/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2008 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# cmd/ssh/ssh-agent/Makefile

PROG= ssh-agent

OBJS    =       \
        ssh-agent.o
SRCS    = $(OBJS:.o=.c)

include ../../Makefile.cmd
include ../Makefile.ssh-common

LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket

# libcrypto has no lint library, so we can only use it when building
$(PROG) := LDLIBS += -lcrypto
LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket -lcrypto

POFILE_DIR= ..

.KEEP_STATE:

.PARALLEL: $(OBJS)

all: $(PROG)

$(PROG): $(OBJS) ../libssh/$(MACH)/libssh.a ../libopenbsd-compat/$(MACH)/libopen
        $(LINK.c) $(OBJS) -o $@ $(LDLIBS) $(DYNFLAGS)
        $(POST_PROCESS)

install: all $(ROOTPROG)

clean:
        $(RM) -f $(OBJS) $(PROG)

lint:   lint_SRCS

include ../Makefile.msg.targ
include ../../Makefile.targ
```

```
**********************************************************
    1551 Thu Feb 13 22:45:21 2014
new/usr/src/cmd/ssh/ssh-keygen/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2008 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# cmd/ssh/ssh-keygen/Makefile

PROG= ssh-keygen

OBJS    =       \
        ssh-keygen.o
SRCS    = $(OBJS:.o=.c)

include ../../Makefile.cmd
include ../Makefile.ssh-common

LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket

# libcrypto has no lint library, so we can only use it when building
$(PROG) := LDLIBS += -lcrypto
LDLIBS += $(SSH_COMMON_LDLIBS) -lcrypto -lsocket

POFILE_DIR= ..

.KEEP_STATE:

.PARALLEL: $(OBJS)

all: $(PROG)

$(PROG): $(OBJS) ../libssh/$(MACH)/libssh.a ../libopenbsd-compat/$(MACH)/libopen
        $(LINK.c) $(OBJS) -o $@ $(LDLIBS) $(DYNFLAGS)
        $(POST_PROCESS)

install: all $(ROOTPROG)

clean:
        $(RM) -f $(OBJS) $(PROG)

lint:   lint_SRCS

include ../Makefile.msg.targ
include ../../Makefile.targ
```

```
**********************************************************
    1574 Thu Feb 13 22:45:21 2014
new/usr/src/cmd/ssh/ssh-keyscan/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2008 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# cmd/ssh/ssh-keyscan/Makefile

PROG= ssh-keyscan

OBJS    =        \
        ssh-keyscan.o
SRCS    = $(OBJS:.o=.c)

include ../../Makefile.cmd
include ../Makefile.ssh-common

LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket -lnsl

# libcrypto and libz have no lint library, so we can only use it when building
$(PROG) := LDLIBS += -lcrypto -lz
LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket -lnsl -lz -lcrypto

POFILE_DIR= ..

.KEEP_STATE:

.PARALLEL: $(OBJS)

all: $(PROG)

$(PROG): $(OBJS) ../libssh/$(MACH)/libssh.a ../libopenbsd-compat/$(MACH)/libopen
        $(LINK.c) $(OBJS) -o $@ $(LDLIBS) $(DYNFLAGS)
        $(POST_PROCESS)

clean:
        $(RM) -f $(OBJS) $(PROG)

lint:   lint_SRCS

include ../Makefile.msg.targ
include ../../Makefile.targ

install: all $(ROOTPROG)
```

```
**********************************************************
    1694 Thu Feb 13 22:45:22 2014
new/usr/src/cmd/ssh/ssh/Makefile
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2008 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# cmd/ssh/ssh/Makefile

PROG= ssh

OBJS    = ssh.o \
          sshconnect.o \
          sshconnect1.o \
          sshconnect2.o \
          sshtty.o \
          clientloop.o \
          gss-clnt.o
SRCS    = $(OBJS:.o=.c)

include ../../Makefile.cmd
include ../Makefile.ssh-common

LDLIBS += $(SSH_COMMON_LDLIBS) -lsocket \
        -lnsl \
        -lz \
        -lcrypto \
        -lgss

# libcrypto and libz have no lint libraries, so we can only use them when
# building
$(PROG) := LDLIBS += -lcrypto -lz

#endif /* ! codereview */
POFILE_DIR= ..

.KEEP_STATE:

.PARALLEL: $(OBJS)

all: $(PROG)

$(PROG): $(OBJS) ../libssh/$(MACH)/libssh.a ../libopenbsd-compat/$(MACH)/libopen
        $(LINK.c) $(OBJS) -o $@ $(LDLIBS) $(DYNFLAGS)
        $(POST_PROCESS)
```

```
install: all $(ROOTPROG)

clean:
        $(RM) -f $(OBJS) $(PROG)

lint:   lint_SRCS

include ../Makefile.msg.targ

XGETFLAGS += --keyword=log
include ../../Makefile.targ
```

```
**********************************************************
    1939 Thu Feb 13 22:45:22 2014
new/usr/src/lib/libkmf/plugins/kmf_openssl/Makefile.com
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# Makefile for KMF Plugins
#

LIBRARY=        kmf_openssl.a
VERS=           .1

OBJECTS=        openssl_spi.o

include $(SRC)/lib/Makefile.lib

LIBLINKS=       $(DYNLIB:.so.1=.so)
KMFINC=         -I../../../include -I../../../ber_der/inc

BERLIB=         -lkmf -lkmfberder
BERLIB64=       $(BERLIB)

OPENSSLLIBS=    $(BERLIB) -lcrypto -lcryptoutil -lc
OPENSSLLIBS64=  $(BERLIB64) -lcrypto -lcryptoutil -lc

LINTSSLLIBS     = $(BERLIB) -lcryptoutil -lc
LINTSSLLIBS64   = $(BERLIB64) -lcryptoutil -lc
LINTSSLLIBS     = $(BERLIB) -lcrypto -lcryptoutil -lc
LINTSSLLIBS64   = $(BERLIB64) -lcrypto -lcryptoutil -lc

SRCDIR=         ../common
INCDIR=         ../../include

CFLAGS          +=      $(CCVERBOSE)
CPPFLAGS        +=      -D_REENTRANT $(KMFINC) \
                        -I$(INCDIR) -I$(ADJUNCT_PROTO)/usr/include/libxml2

CERRWARN        +=      -_gcc=-Wno-unused-label
CERRWARN        +=      -_gcc=-Wno-unused-value
CERRWARN        +=      -_gcc=-Wno-uninitialized

PICS=   $(OBJECTS:%=pics/%)

lint:=  OPENSSLLIBS=    $(LINTSSLLIBS)
```

```
lint:=  OPENSSLLIBS64=  $(LINTSSLLIBS64)

LDLIBS32        +=      $(OPENSSLLIBS)

ROOTLIBDIR=     $(ROOTFS_LIBDIR)/crypto
ROOTLIBDIR64=   $(ROOTFS_LIBDIR)/crypto/$(MACH64)

.KEEP_STATE:

LIBS    =       $(DYNLIB)
all:    $(DYNLIB) $(LINTLIB)

lint: lintcheck

FRC:

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

```
**********************************************************
  133359 Thu Feb 13 22:45:23 2014
new/usr/src/lib/libkmf/plugins/kmf_openssl/common/openssl_spi.c
(void) an sk_set via a macro which upsets lint.  If the macro is
complicated, this might need to be /* LINTED */ instead
remove lint supression made unnecessary by the openssl upgrade
**********************************************************
______unchanged_portion_omitted_

/* ocsp_find_signer_sk() is copied from openssl source */
static X509 *ocsp_find_signer_sk(STACK_OF(X509) *certs, OCSP_RESPID *id)
{
        int i;
        unsigned char tmphash[SHA_DIGEST_LENGTH], *keyhash;

        /* Easy if lookup by name */
        if (id->type == V_OCSP_RESPID_NAME)
                return (X509_find_by_subject(certs, id->value.byName));

        /* Lookup by key hash */

        /* If key hash isn’t SHA1 length then forget it */
        if (id->value.byKey->length != SHA_DIGEST_LENGTH)
                return (NULL);

        keyhash = id->value.byKey->data;
        /* Calculate hash of each key and compare */
        for (i = 0; i < sk_X509_num(certs); i++) {
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                X509 *x = sk_X509_value(certs, i);
                /* Use pubkey_digest to get the key ID value */
                (void) X509_pubkey_digest(x, EVP_sha1(), tmphash, NULL);
                if (!memcmp(keyhash, tmphash, SHA_DIGEST_LENGTH))
                        return (x);
        }
        return (NULL);
}
______unchanged_portion_omitted_

/*
 * Helper function to extract keys and certificates from
 * a single PEM file.  Typically the file should contain a
 * private key and an associated public key wrapped in an x509 cert.
 * However, the file may be just a list of X509 certs with no keys.
 */
static KMF_RETURN
extract_pem(KMF_HANDLE *kmfh,
        char *issuer, char *subject, KMF_BIGINT *serial,
        char *filename, CK_UTF8CHAR *pin,
        CK_ULONG pinlen, EVP_PKEY **priv_key, KMF_DATA **certs,
        int *numcerts)
/* ARGSUSED6 */
{
        KMF_RETURN rv = KMF_OK;
        FILE *fp;
        STACK_OF(X509_INFO) *x509_info_stack = NULL;
        int i, ncerts = 0, matchcerts = 0;
        EVP_PKEY *pkey = NULL;
        X509_INFO *info;
        X509 *x;
        X509_INFO **cert_infos = NULL;
        KMF_DATA *certlist = NULL;

        if (priv_key)
                *priv_key = NULL;
        if (certs)
```

```
                *certs = NULL;
        fp = fopen(filename, "r");
        if (fp == NULL)
                return (KMF_ERR_OPEN_FILE);

        x509_info_stack = PEM_X509_INFO_read(fp, NULL, NULL, pin);
        if (x509_info_stack == NULL) {
                (void) fclose(fp);
                return (KMF_ERR_ENCODING);
        }
        cert_infos = (X509_INFO **)malloc(sk_X509_INFO_num(x509_info_stack) *
            sizeof (X509_INFO *));
        if (cert_infos == NULL) {
                (void) fclose(fp);
                rv = KMF_ERR_MEMORY;
                goto err;
        }

        for (i = 0; i < sk_X509_INFO_num(x509_info_stack); i++) {
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                cert_infos[ncerts] = sk_X509_INFO_value(x509_info_stack, i);
                ncerts++;
        }

        if (ncerts == 0) {
                (void) fclose(fp);
                rv = KMF_ERR_CERT_NOT_FOUND;
                goto err;
        }

        if (priv_key != NULL) {
                rewind(fp);
                pkey = PEM_read_PrivateKey(fp, NULL, NULL, pin);
        }
        (void) fclose(fp);

        x = cert_infos[ncerts - 1]->x509;
        /*
         * Make sure the private key matchs the last cert in the file.
         */
        if (pkey != NULL && !X509_check_private_key(x, pkey)) {
                EVP_PKEY_free(pkey);
                rv = KMF_ERR_KEY_MISMATCH;
                goto err;
        }

        certlist = (KMF_DATA *)calloc(ncerts, sizeof (KMF_DATA));
        if (certlist == NULL) {
                if (pkey != NULL)
                        EVP_PKEY_free(pkey);
                rv = KMF_ERR_MEMORY;
                goto err;
        }

        /*
         * Convert all of the certs to DER format.
         */
        matchcerts = 0;
        for (i = 0; rv == KMF_OK && certs != NULL && i < ncerts; i++) {
                boolean_t match = FALSE;
                info =  cert_infos[ncerts - 1 - i];

                rv = check_cert(info->x509, issuer, subject, serial, &match);
                if (rv != KMF_OK || match != TRUE) {
                        rv = KMF_OK;
                        continue;
                }

                rv = ssl_cert2KMFDATA(kmfh, info->x509,
                        &certlist[matchcerts++]);

                if (rv != KMF_OK) {
                        int j;
                        for (j = 0; j < matchcerts; j++)
                                kmf_free_data(&certlist[j]);
                        free(certlist);
                        certlist = NULL;
                        ncerts = matchcerts = 0;
                }
        }

        if (numcerts != NULL)
                *numcerts = matchcerts;

        if (certs != NULL)
                *certs = certlist;
        else if (certlist != NULL) {
                for (i = 0; i < ncerts; i++)
                        kmf_free_data(&certlist[i]);
                free(certlist);
                certlist = NULL;
        }

        if (priv_key == NULL && pkey != NULL)
                EVP_PKEY_free(pkey);
        else if (priv_key != NULL && pkey != NULL)
                *priv_key = pkey;

err:
        /* Cleanup the stack of X509 info records */
        for (i = 0; i < sk_X509_INFO_num(x509_info_stack); i++) {
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                info = (X509_INFO *)sk_X509_INFO_value(x509_info_stack, i);
                X509_INFO_free(info);
        }
        if (x509_info_stack)
                sk_X509_INFO_free(x509_info_stack);

        if (cert_infos != NULL)
                free(cert_infos);

        return (rv);
}

static KMF_RETURN
openssl_parse_bags(STACK_OF(PKCS12_SAFEBAG) *bags, char *pin,
        STACK_OF(EVP_PKEY) *keys, STACK_OF(X509) *certs)
{
        KMF_RETURN ret;
        int i;

        for (i = 0; i < sk_PKCS12_SAFEBAG_num(bags); i++) {
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                PKCS12_SAFEBAG *bag = sk_PKCS12_SAFEBAG_value(bags, i);
                ret = openssl_parse_bag(bag, pin, (pin ? strlen(pin) : 0),
                    keys, certs);

                if (ret != KMF_OK)
                        return (ret);
        }

        return (ret);
```

```
}

static KMF_RETURN
set_pkey_attrib(EVP_PKEY *pkey, ASN1_TYPE *attrib, int nid)
{
        X509_ATTRIBUTE *attr = NULL;

        if (pkey == NULL || attrib == NULL)
                return (KMF_ERR_BAD_PARAMETER);

        if (pkey->attributes == NULL) {
                pkey->attributes = sk_X509_ATTRIBUTE_new_null();
                if (pkey->attributes == NULL)
                        return (KMF_ERR_MEMORY);
        }
        attr = X509_ATTRIBUTE_create(nid, attrib->type, attrib->value.ptr);
        if (attr != NULL) {
                int i;
                X509_ATTRIBUTE *a;
                for (i = 0;
                    i < sk_X509_ATTRIBUTE_num(pkey->attributes); i++) {
                        /* LINTED E_BAD_PTR_CASE_ALIGN */
                        a = sk_X509_ATTRIBUTE_value(pkey->attributes, i);
                        if (OBJ_obj2nid(a->object) == nid) {
                                X509_ATTRIBUTE_free(a);
                                (void) sk_X509_ATTRIBUTE_set(pkey->attributes,
                                /* LINTED E_BAD_PTR_CAST_ALIGN */
                                sk_X509_ATTRIBUTE_set(pkey->attributes,
                                    i, attr);
                                return (KMF_OK);
                        }
                }
                if (sk_X509_ATTRIBUTE_push(pkey->attributes, attr) == NULL) {
                        X509_ATTRIBUTE_free(attr);
                        return (KMF_ERR_MEMORY);
                }
        } else {
                return (KMF_ERR_MEMORY);
        }

        return (KMF_OK);
}
______unchanged_portion_omitted_

static KMF_RETURN
openssl_pkcs12_parse(PKCS12 *p12, char *pin,
        STACK_OF(EVP_PKEY) *keys,
        STACK_OF(X509) *certs,
        STACK_OF(X509) *ca)
/* ARGSUSED3 */
{
        KMF_RETURN ret = KMF_OK;
        STACK_OF(PKCS7) *asafes = NULL;
        STACK_OF(PKCS12_SAFEBAG) *bags = NULL;
        int i, bagnid;
        PKCS7 *p7;

        if (p12 == NULL || (keys == NULL && certs == NULL))
                return (KMF_ERR_BAD_PARAMETER);

        if (pin == NULL || *pin == NULL) {
                if (PKCS12_verify_mac(p12, NULL, 0)) {
                        pin = NULL;
                } else if (PKCS12_verify_mac(p12, "", 0)) {
                        pin = "";
                } else {
                        return (KMF_ERR_AUTH_FAILED);
                }
        } else if (!PKCS12_verify_mac(p12, pin, -1)) {
                return (KMF_ERR_AUTH_FAILED);
        }

        if ((asafes = PKCS12_unpack_authsafes(p12)) == NULL)
                return (KMF_ERR_PKCS12_FORMAT);

        for (i = 0; ret == KMF_OK && i < sk_PKCS7_num(asafes); i++) {
                bags = NULL;
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                p7 = sk_PKCS7_value(asafes, i);
                bagnid = OBJ_obj2nid(p7->type);

                if (bagnid == NID_pkcs7_data) {
                        bags = PKCS12_unpack_p7data(p7);
                } else if (bagnid == NID_pkcs7_encrypted) {
                        bags = PKCS12_unpack_p7encdata(p7, pin,
                            (pin ? strlen(pin) : 0));
                } else {
                        continue;
                }
                if (bags == NULL) {
                        ret = KMF_ERR_PKCS12_FORMAT;
                        goto out;
                }

                if (openssl_parse_bags(bags, pin, keys, certs) != KMF_OK)
                        ret = KMF_ERR_PKCS12_FORMAT;

                sk_PKCS12_SAFEBAG_pop_free(bags, PKCS12_SAFEBAG_free);
        }
out:
        if (asafes != NULL)
                sk_PKCS7_pop_free(asafes, PKCS7_free);

        return (ret);
}
```

______***unchanged_portion_omitted_***

```
static X509_ATTRIBUTE *
find_attr(STACK_OF(X509_ATTRIBUTE) *attrs, int nid)
{
        X509_ATTRIBUTE *a;
        int i;

        if (attrs == NULL)
                return (NULL);

        for (i = 0; i < sk_X509_ATTRIBUTE_num(attrs); i++) {
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                a = sk_X509_ATTRIBUTE_value(attrs, i);
                if (OBJ_obj2nid(a->object) == nid)
                        return (a);
        }
        return (NULL);
}

static KMF_RETURN
convertToRawKey(EVP_PKEY *pkey, KMF_RAW_KEY_DATA *key)
{
        KMF_RETURN rv = KMF_OK;
        X509_ATTRIBUTE *attr;

        if (pkey == NULL || key == NULL)
```

```
                return (KMF_ERR_BAD_PARAMETER);
        /* Convert SSL key to raw key */
        switch (pkey->type) {
                case EVP_PKEY_RSA:
                        rv = exportRawRSAKey(EVP_PKEY_get1_RSA(pkey),
                            key);
                        if (rv != KMF_OK)
                                return (rv);
                        break;
                case EVP_PKEY_DSA:
                        rv = exportRawDSAKey(EVP_PKEY_get1_DSA(pkey),
                            key);
                        if (rv != KMF_OK)
                                return (rv);
                        break;
                default:
                        return (KMF_ERR_BAD_PARAMETER);
        }
        /*
         * If friendlyName, add it to record.
         */
        attr = find_attr(pkey->attributes, NID_friendlyName);
        if (attr != NULL) {
                ASN1_TYPE *ty = NULL;
                int numattr = sk_ASN1_TYPE_num(attr->value.set);
                if (attr->single == 0 && numattr > 0) {
                        /* LINTED E_BAD_PTR_CAST_ALIGN */
                        ty = sk_ASN1_TYPE_value(attr->value.set, 0);
                }
                if (ty != NULL) {
#if OPENSSL_VERSION_NUMBER < 0x10000000L
                        key->label = uni2asc(ty->value.bmpstring->data,
                            ty->value.bmpstring->length);
#else
                        key->label = OPENSSL_uni2asc(ty->value.bmpstring->data,
                            ty->value.bmpstring->length);
#endif
                }
        } else {
                key->label = NULL;
        }

        /*
         * If KeyID, add it to record as a KMF_DATA object.
         */
        attr = find_attr(pkey->attributes, NID_localKeyID);
        if (attr != NULL) {
                ASN1_TYPE *ty = NULL;
                int numattr = sk_ASN1_TYPE_num(attr->value.set);
                if (attr->single == 0 && numattr > 0) {
                        /* LINTED E_BAD_PTR_CAST_ALIGN */
                        ty = sk_ASN1_TYPE_value(attr->value.set, 0);
                }
                key->id.Data = (uchar_t *)malloc(
                    ty->value.octet_string->length);
                if (key->id.Data == NULL)
                        return (KMF_ERR_MEMORY);
                (void) memcpy(key->id.Data, ty->value.octet_string->data,
                    ty->value.octet_string->length);
                key->id.Length = ty->value.octet_string->length;
        } else {
                (void) memset(&key->id, 0, sizeof (KMF_DATA));
        }

        return (rv);
}
```

```
static KMF_RETURN
convertPK12Objects(
        KMF_HANDLE *kmfh,
        STACK_OF(EVP_PKEY) *sslkeys,
        STACK_OF(X509) *sslcert,
        STACK_OF(X509) *sslcacerts,
        KMF_RAW_KEY_DATA **keylist, int *nkeys,
        KMF_X509_DER_CERT **certlist, int *ncerts)
{
        KMF_RETURN rv = KMF_OK;
        KMF_RAW_KEY_DATA key;
        int i;

        for (i = 0; sslkeys != NULL && i < sk_EVP_PKEY_num(sslkeys); i++) {
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                EVP_PKEY *pkey = sk_EVP_PKEY_value(sslkeys, i);
                rv = convertToRawKey(pkey, &key);
                if (rv == KMF_OK)
                        rv = add_key_to_list(keylist, &key, nkeys);

                if (rv != KMF_OK)
                        return (rv);
        }

        /* Now add the certificate to the certlist */
        for (i = 0; sslcert != NULL && i < sk_X509_num(sslcert); i++) {
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                X509 *cert = sk_X509_value(sslcert, i);
                rv = add_cert_to_list(kmfh, cert, certlist, ncerts);
                if (rv != KMF_OK)
                        return (rv);
        }

        /* Also add any included CA certs to the list */
        for (i = 0; sslcacerts != NULL && i < sk_X509_num(sslcacerts); i++) {
                X509 *c;
                /*
                 * sk_X509_value() is macro that embeds a cast to (X509 *).
                 * Here it translates into ((X509 *)sk_value((ca), (i))).
                 * Lint is complaining about the embedded casting, and
                 * to fix it, you need to fix openssl header files.
                 */
                /* LINTED E_BAD_PTR_CAST_ALIGN */
                c = sk_X509_value(sslcacerts, i);

                /* Now add the ca cert to the certlist */
                rv = add_cert_to_list(kmfh, c, certlist, ncerts);
                if (rv != KMF_OK)
                        return (rv);
        }
        return (rv);
}
```

______unchanged_portion_omitted_

```
KMF_RETURN
OpenSSL_FindCertInCRL(KMF_HANDLE_T handle, int numattr, KMF_ATTRIBUTE *attrlist)
{
        KMF_RETURN ret = KMF_OK;
        KMF_HANDLE *kmfh = (KMF_HANDLE *)handle;
        KMF_ENCODE_FORMAT format;
        BIO *in = NULL;
        X509   *xcert = NULL;
        X509_CRL   *xcrl = NULL;
        STACK_OF(X509_REVOKED) *revoke_stack = NULL;
        X509_REVOKED *revoke;
```

```
        int i;
        char *crlfilename, *crlfile, *dirpath, *certfile;

        if (numattr == 0 || attrlist == NULL) {
                return (KMF_ERR_BAD_PARAMETER);
        }

        crlfilename = kmf_get_attr_ptr(KMF_CRL_FILENAME_ATTR,
            attrlist, numattr);

        if (crlfilename == NULL)
                return (KMF_ERR_BAD_CRLFILE);

        certfile = kmf_get_attr_ptr(KMF_CERT_FILENAME_ATTR, attrlist, numattr);
        if (certfile == NULL)
                return (KMF_ERR_BAD_CRLFILE);

        dirpath = kmf_get_attr_ptr(KMF_DIRPATH_ATTR, attrlist, numattr);

        crlfile = get_fullpath(dirpath, crlfilename);

        if (crlfile == NULL)
                return (KMF_ERR_BAD_CRLFILE);

        if (isdir(crlfile)) {
                ret = KMF_ERR_BAD_CRLFILE;
                goto end;
        }

        ret = kmf_is_crl_file(handle, crlfile, &format);
        if (ret != KMF_OK)
                goto end;

        /* Read the CRL file and load it into a X509_CRL structure */
        in = BIO_new_file(crlfilename, "rb");
        if (in == NULL) {
                SET_ERROR(kmfh, ERR_get_error());
                ret = KMF_ERR_OPEN_FILE;
                goto end;
        }

        if (format == KMF_FORMAT_ASN1) {
                xcrl = d2i_X509_CRL_bio(in, NULL);
        } else if (format == KMF_FORMAT_PEM) {
                xcrl = PEM_read_bio_X509_CRL(in, NULL, NULL, NULL);
        }

        if (xcrl == NULL) {
                SET_ERROR(kmfh, ERR_get_error());
                ret = KMF_ERR_BAD_CRLFILE;
                goto end;
        }
        (void) BIO_free(in);

        /* Read the Certificate file and load it into a X509 structure */
        ret = kmf_is_cert_file(handle, certfile, &format);
        if (ret != KMF_OK)
                goto end;

        in = BIO_new_file(certfile, "rb");
        if (in == NULL) {
                SET_ERROR(kmfh, ERR_get_error());
                ret = KMF_ERR_OPEN_FILE;
                goto end;
        }
```

```
	if (format == KMF_FORMAT_ASN1) {
		xcert = d2i_X509_bio(in, NULL);
	} else if (format == KMF_FORMAT_PEM) {
		xcert = PEM_read_bio_X509(in, NULL, NULL, NULL);
	}

	if (xcert == NULL) {
		SET_ERROR(kmfh, ERR_get_error());
		ret = KMF_ERR_BAD_CERTFILE;
		goto end;
	}

	/* Check if the certificate and the CRL have same issuer */
	if (X509_NAME_cmp(xcert->cert_info->issuer, xcrl->crl->issuer) != 0) {
		ret = KMF_ERR_ISSUER;
		goto end;
	}

	/* Check to see if the certificate serial number is revoked */
	revoke_stack = X509_CRL_get_REVOKED(xcrl);
	if (sk_X509_REVOKED_num(revoke_stack) <= 0) {
		/* No revoked certificates in the CRL file */
		SET_ERROR(kmfh, ERR_get_error());
		ret = KMF_ERR_EMPTY_CRL;
		goto end;
	}

	for (i = 0; i < sk_X509_REVOKED_num(revoke_stack); i++) {
		/* LINTED E_BAD_PTR_CAST_ALIGN */
		revoke = sk_X509_REVOKED_value(revoke_stack, i);
		if (ASN1_INTEGER_cmp(xcert->cert_info->serialNumber,
		    revoke->serialNumber) == 0) {
			break;
		}
	}

	if (i < sk_X509_REVOKED_num(revoke_stack)) {
		ret = KMF_OK;
	} else {
		ret = KMF_ERR_NOT_REVOKED;
	}

end:
	if (in != NULL)
		(void) BIO_free(in);
	if (xcrl != NULL)
		X509_CRL_free(xcrl);
	if (xcert != NULL)
		X509_free(xcert);

	return (ret);
}
```

______unchanged_portion_omitted_

```
**********************************************************
    2531 Thu Feb 13 22:45:23 2014
new/usr/src/lib/libpkg/Makefile.com
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#

#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

LIBRARY= libpkg.a
VERS=   .1

# include library definitions
OBJECTS=        \
                canonize.o   ckparam.o    ckvolseq.o \
                devtype.o    dstream.o    gpkglist.o \
                gpkgmap.o    isdir.o      logerr.o \
                mappath.o    ncgrpw.o     nhash.o \
                pkgexecl.o   pkgexecv.o   pkgmount.o \
                pkgtrans.o   ppkgmap.o \
                progerr.o    putcfile.o   rrmdir.o \
                runcmd.o     srchcfile.o  tputcfent.o \
                verify.o     security.o   pkgweb.o \
                pkgerr.o     keystore.o   p12lib.o \
                vfpops.o     fmkdir.o     pkgstr.o \
                handlelocalfs.o pkgserv.o


# include library definitions
include $(SRC)/lib/Makefile.lib

SRCDIR=         ../common

POFILE =        libpkg.po
MSGFILES =      $(OBJECTS:%.o=../common/%.i)
CLEANFILES +=   $(MSGFILES)

# This library is NOT lint clean

# openssl forces us to ignore dubious pointer casts, thanks to its clever
# use of macros for stack management.
LINTFLAGS=      -umx -errtags \
                -erroff=E_BAD_PTR_CAST_ALIGN,E_BAD_PTR_CAST
$(LINTLIB):=    SRCS = $(SRCDIR)/$(LINTSRC)
```

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


LDLIBS +=       -lc -lwanboot -lscf -ladm

# libcrypto and libssl have no lint library, and so can only be used when
# building
$(DYNLIB) := -lcrypto -lssl
LDLIBS +=       -lc -lssl -lwanboot -lcrypto -lscf -ladm

CFLAGS +=       $(CCVERBOSE)
CERRWARN +=     -_gcc=-Wno-unused-label
CERRWARN +=     -_gcc=-Wno-parentheses
CERRWARN +=     -_gcc=-Wno-uninitialized
CERRWARN +=     -_gcc=-Wno-clobbered
CERRWARN +=     -_gcc=-Wno-switch
CERRWARN +=     -_gcc=-Wno-unused-value
CPPFLAGS +=     -I$(SRCDIR) -D_FILE_OFFSET_BITS=64

.KEEP_STATE:

all:    $(LIBS)

$(POFILE): $(MSGFILES)
        $(BUILDPO.msgfiles)

_msg: $(MSGDOMAINPOFILE)

lint: lintcheck

# include library targets
include $(SRC)/lib/Makefile.targ
include $(SRC)/Makefile.msg.targ
```

```
**********************************************************
    1904 Thu Feb 13 22:45:24 2014
new/usr/src/lib/libshare/Makefile.com
don’t pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#
LIBRARY =       libshare.a
VERS =          .1
NFSLIB_DIR =    $(SRC)/cmd/fs.d/nfs/lib

LIBOBJS =       libshare.o libsharecore.o scfutil.o libshare_zfs.o \
                plugin.o parser.o issubdir.o
OTHOBJS =       sharetab.o nfs_sec.o
OBJECTS =       $(LIBOBJS) $(OTHOBJS)

include ../../Makefile.lib
SRCDIR =        ../common

LIBSRCS =       $(LIBOBJS:%.o=$(SRCDIR)/%.c)
# we don’t want to lint the sharetab and nfs_sec files
lintcheck := SRCS = $(LIBSRCS)

LIBS =          $(DYNLIB) $(LINTLIB)
LDLIBS +=       -lc -lnsl -lscf -lzfs -luuid -lnvpair
LDLIBS +=       -lc -lnsl -lscf -lzfs -luuid -lxml2 -lnvpair
$(LINTLIB) :=   SRCS = $(SRCDIR)/$(LINTSRC)

# libxml2 has no lint library, so we can only use it when building
$(DYNLIB) := LDLIBS += -lxml2

#endif /* ! codereview */
#add nfs/lib directory as part of the include path
CFLAGS +=       $(CCVERBOSE)
CERRWARN +=     -_gcc=-Wno-parentheses
CERRWARN +=     -_gcc=-Wno-uninitialized
CERRWARN +=     -_gcc=-Wno-switch
CPPFLAGS +=     -D_REENTRANT -I$(NFSLIB_DIR) \
                -I$(ADJUNCT_PROTO)/usr/include/libxml2

.KEEP_STATE:

all: $(LIBS)

lint: lintcheck
```

```
pics/%.o:       $(NFSLIB_DIR)/%.c
        $(COMPILE.c) -o $@ $<
        $(POST_PROCESS_O)

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

```
**********************************************************
    2055 Thu Feb 13 22:45:24 2014
new/usr/src/lib/libshare/autofs/Makefile.com
don’t pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
# Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
#
#

LIBRARY =       libshare_autofs.a
VERS =          .1
AUTOFSSMFLIB_DIR = $(SRC)/cmd/fs.d/nfs/lib

LIBOBJS =       libshare_autofs.o
OTHOBJS =       smfcfg.o
OBJECTS =       $(LIBOBJS) $(OTHOBJS)

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

ROOTLIBDIR =    $(ROOT)/usr/lib/fs/autofs
ROOTLIBDIR64 =  $(ROOT)/usr/lib/fs/autofs/$(MACH64)

LIBSRCS = $(LIBOBJS:%.o=$(SRCDIR)/%.c)
# we don’t want to lint the sources for OTHOBJS since they are pre-existing file
# that are not lint free.
lintcheck := SRCS = $(LIBSRCS)

LIBS =          $(DYNLIB)
LDLIBS +=       -lshare -lscf -lumem -lc

# libxml2 has no lint library, so we can only use it when building
$(DYNLIB) := LDLIBS += -lxml2
LDLIBS +=       -lshare -lscf -lumem -lc -lxml2

#add nfs/lib directory as part of the include path
CFLAGS +=       $(CCVERBOSE)
CERRWARN +=     -_gcc=-Wno-switch
CERRWARN +=     -_gcc=-Wno-unused-variable
CERRWARN +=     -_gcc=-Wno-uninitialized
CPPFLAGS +=     -D_REENTRANT -I$(AUTOFSSMFLIB_DIR) \
                        -I$(ADJUNCT_PROTO)/usr/include/libxml2 \
                        -I$(SRCDIR)../common

.KEEP_STATE:
```

```
all: $(LIBS)

install: $(ROOTLIBDIR) $(ROOTLIBDIR64) all

lint: lintcheck

pics/%.o:       $(AUTOFSSMFLIB_DIR)/%.c
        $(COMPILE.c) -o $@ $<
        $(POST_PROCESS_O)

$(ROOTLIBDIR):
        $(INS.dir)

$(ROOTLIBDIR64):
        $(INS.dir)

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

```
**********************************************************
    2011 Thu Feb 13 22:45:25 2014
new/usr/src/lib/libshare/nfs/Makefile.com
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
#
# Copyright (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
#

LIBRARY =       libshare_nfs.a
VERS =          .1
NFSLIB_DIR      = $(SRC)/cmd/fs.d/nfs/lib

LIBOBJS =       libshare_nfs.o
OTHOBJS =       nfs_sec.o nfslog_config.o nfslogtab.o smfcfg.o
OBJECTS =       $(LIBOBJS) $(OTHOBJS)

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

ROOTLIBDIR =    $(ROOT)/usr/lib/fs/nfs
ROOTLIBDIR64 =  $(ROOT)/usr/lib/fs/nfs/$(MACH64)

LIBSRCS = $(LIBOBJS:%.o=$(SRCDIR)/%.c)
# we don't want to lint the sources for OTHOBJS since they are pre-existing file
# that are not lint free.
lintcheck := SRCS = $(LIBSRCS)

LIBS =          $(DYNLIB)
LDLIBS +=       -lshare -lnsl -lscf -lumem -lc

# libxml2 has no lint library, so we can only use it when building
$(DYNLIB) := LDLIBS += -lxml2
LDLIBS +=       -lshare -lnsl -lscf -lumem -lc -lxml2

#add nfs/lib directory as part of the include path
CFLAGS +=       $(CCVERBOSE)
CERRWARN +=     -_gcc=-Wno-parentheses
CERRWARN +=     -_gcc=-Wno-switch
CERRWARN +=     -_gcc=-Wno-unused-variable
CERRWARN +=     -_gcc=-Wno-uninitialized
CPPFLAGS +=     -D_REENTRANT -I$(NFSLIB_DIR) \
                -I$(ADJUNCT_PROTO)/usr/include/libxml2 -I$(SRCDIR)/../common

.KEEP_STATE:
```

```
all: $(LIBS)

install: all

lint: lintcheck

pics/%.o:  $(NFSLIB_DIR)/%.c
        $(COMPILE.c) -o $@ $<
        $(POST_PROCESS_O)

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

```
**********************************************************
    1762 Thu Feb 13 22:45:25 2014
new/usr/src/lib/libshare/smbfs/Makefile.com
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
#
# Copyright 2008 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#

LIBRARY =       libshare_smbfs.a
VERS =          .1

LIBOBJS =       libshare_smbfs.o
SMBBASE_OBJ =   smbfs_scfutil.o
OBJECTS =       $(LIBOBJS) $(SMBBASE_OBJ)

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

ROOTLIBDIR =    $(ROOT)/usr/lib/fs/smbfs
ROOTLIBDIR64 =  $(ROOT)/usr/lib/fs/smbfs/$(MACH64)

LIBSRCS = $(LIBOBJS:%.o=$(SRCDIR)/%.c)

LIBS =          $(DYNLIB)
LDLIBS +=       -lshare -lscf -lumem -luuid -lc -lsmbfs

# libxml2 has no lint library, so we can only use it when building
$(DYNLIB) := LDLIBS += -lxml2
LDLIBS +=       -lshare -lscf -lumem -luuid -lc -lxml2 -lsmbfs

CFLAGS +=       $(CCVERBOSE)
CERRWARN +=     -_gcc=-Wno-switch
CERRWARN +=     -_gcc=-Wno-uninitialized
CPPFLAGS +=     -D_REENTRANT -I$(ADJUNCT_PROTO)/usr/include/libxml2 \
                -I$(SRCDIR)/../common -I$(SRC)/lib/libsmbfs -I$(SRC)/uts/common

.KEEP_STATE:

all: $(LIBS)

install: $(ROOTLIBDIR) $(ROOTLIBDIR64) all

lint: lintcheck

$(ROOTLIBDIR):
```

```
        $(INS.dir)

$(ROOTLIBDIR64):
        $(INS.dir)

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

****
2761 Thu Feb 13 22:45:25 2014
new/usr/src/lib/libwanboot/Makefile.com
don't pass in lint libraries which no longer exist
****

```
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
# Copyright (c) 2012 by Delphix. All rights reserved.
#

LIBRARY =       libwanboot.a
VERS =          .1

# List of locally located modules.
LOC_DIR =       ../common
LOC_OBJS =      socket_inet.o bootinfo_aux.o
LOC_SRCS =      $(LOC_OBJS:%.o=$(LOC_DIR)/%.c)

# List of common wanboot objects.
COM_DIR =       ../../../common/net/wanboot
COM_OBJS =      auxutil.o \
                boot_http.o \
                bootconf.o \
                bootconf_errmsg.o \
                bootinfo.o \
                bootlog.o \
                http_errorstr.o \
                p12access.o \
                p12auxpars.o \
                p12auxutl.o \
                p12err.o \
                p12misc.o \
                parseURL.o
COM_SRCS =      $(COM_OBJS:%.o=$(COM_DIR)/%.c)

# List of common DHCP modules.
DHCP_DIR =      $(SRC)/common/net/dhcp
DHCP_OBJS =     dhcpinfo.o
DHCP_SRCS =     $(DHCP_OBJS:%.o=$(DHCP_DIR)/%.c)

OBJECTS =       $(LOC_OBJS) $(COM_OBJS) $(DHCP_OBJS)

include ../../Makefile.lib

LIBS +=         $(LINTLIB)
```

```
LDLIBS +=       -lnvpair -lresolv -lnsl -lsocket -ldevinfo -ldhcputil \
                -linetutil -lc

# libcrypto and libssl have no lint library, so we can only use it when
# building
$(DYNLIB) := LDLIBS += -lcrypto -lssl

                -linetutil -lc -lcrypto -lssl
CPPFLAGS =      -I$(SRC)/common/net/wanboot/crypt $(CPPFLAGS.master)
CERRWARN +=     -_gcc=-Wno-switch
CERRWARN +=     -_gcc=-Wno-parentheses
CERRWARN +=     -_gcc=-Wno-unused-value
CERRWARN +=     -_gcc=-Wno-uninitialized

# Must override SRCS from Makefile.lib since sources have
# multiple source directories.
SRCS =          $(LOC_SRCS) $(COM_SRCS) $(DHCP_SRCS)

# Must define location of lint library source.
SRCDIR =        $(LOC_DIR)
$(LINTLIB) :=   SRCS = $(SRCDIR)/$(LINTSRC)

# OpenSSL requires us to turn this off
LINTFLAGS   +=    -erroff=E_BAD_PTR_CAST_ALIGN
LINTFLAGS64 +=    -erroff=E_BAD_PTR_CAST_ALIGN

CFLAGS +=       $(CCVERBOSE)
CPPFLAGS +=     -I$(LOC_DIR) -I$(COM_DIR) -I$(DHCP_DIR)

.KEEP_STATE:

all: $(LIBS)

lint: lintcheck

pics/%.o: $(COM_DIR)/%.c
        $(COMPILE.c) -o $@ $<
        $(POST_PROCESS_O)

pics/%.o: $(DHCP_DIR)/%.c
        $(COMPILE.c) -o $@ $<
        $(POST_PROCESS_O)

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

```
**********************************************************
    2467 Thu Feb 13 22:45:26 2014
new/usr/src/lib/pkcs11/pkcs11_tpm/Makefile.com
don't pass in lint libraries which no longer exist
**********************************************************
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
#
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
# Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
LIBRARY =       pkcs11_tpm.a
VERS =          .1

OBJECTS= api_interface.o \
        apiutil.o \
        asn1.o \
        cert.o \
        data_obj.o \
        decr_mgr.o \
        dig_mgr.o \
        encr_mgr.o \
        globals.o \
        hwf_obj.o \
        key.o \
        key_mgr.o \
        loadsave.o \
        log.o \
        mech_md5.o \
        mech_rsa.o \
        mech_sha.o \
        new_host.o \
        obj_mgr.o \
        object.o \
        sess_mgr.o \
        sign_mgr.o \
        template.o \
        tpm_specific.o \
        utility.o \
        verify_mgr.o


include $(SRC)/lib/Makefile.lib

SRCDIR= ../common

SRCS=   $(OBJECTS:%.o=$(SRCDIR)/%.c)

#       set signing mode
```

```
POST_PROCESS_SO +=      ; $(ELFSIGN_CRYPTO)

ROOTLIBDIR=$(ROOT)/usr/lib/security
ROOTLIBDIR64=$(ROOT)/usr/lib/security/$(MACH64)

LIBS=$(DYNLIB) $(DYNLIB64)

TSSROOT=$(ADJUNCT_PROTO)
TSPILIBDIR=$(TSSROOT)/usr/lib
TSPIINCDIR=$(TSSROOT)/usr/include
TSSLIB=-L$(TSPILIBDIR)
TSSLIB64=-L$(TSPILIBDIR)/$(MACH64)
TSSINC=-I$(TSPIINCDIR)

LDLIBS += $(TSSLIB) -L$(ADJUNCT_PROTO)/lib -lc -luuid -lmd -ltspi -lcrypto

# libtspi and libcrypto have no lint library, so we can only use it when
# building
$(LIBS) := LDLIBS += -ltspi -lcrypto

#endif /* ! codereview */
CPPFLAGS += -xCC -D_POSIX_PTHREAD_SEMANTICS $(TSSINC)
CPPFLAGS64 += $(CPPFLAGS)
C99MODE=        $(C99_ENABLE)

CERRWARN +=     -_gcc=-Wno-parentheses
CERRWARN +=     -_gcc=-Wno-unused-label
CERRWARN +=     -_gcc=-Wno-uninitialized

LINTSRC= $(OBJECTS:%.o=$(SRCDIR)/%.c)

$(LINTLIB):=    SRCS    =       $(SRCDIR)/$(LINTSRC)
LINTSRC= $(SRCS)

CLOBBERFILES += C.ln

.KEEP_STATE:

all: $(LIBS)

lint: $$(LINTSRC)
        $(LINT.c) $(LINTCHECKFLAGS) $(LINTSRC) $(LDLIBS)

pics/%.o: $(SRCDIR)/%.c
        $(COMPILE.c) -o $@ $<
        $(POST_PROCESS_O)

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

```
**********************************************************
   12578 Thu Feb 13 22:45:26 2014
new/usr/src/man/man1/Makefile
remove machid manual pages, since machid was removed
**********************************************************
#
# This file and its contents are supplied under the terms of the
# Common Development and Distribution License ("CDDL"), version 1.0.
# You may only use this file in accordance with the terms of version
# 1.0 of the CDDL.
#
# A full copy of the text of the CDDL should have accompanied this
# source.  A copy of the CDDL is also available via the Internet
# at http://www.illumos.org/license/CDDL.
#

#
# Copyright 2011, Richard Lowe
# Copyright 2012 OmniTI Computer Consulting, Inc.  All rights reserved.
# Copyright 2013 Nexenta Systems, Inc.  All rights reserved.
#

include         $(SRC)/Makefile.master

MANSECT=        1

MANFILES=       acctcom.1                       \
                adb.1                           \
                addbib.1                        \
                alias.1                         \
                allocate.1                      \
                amt.1                           \
                appcert.1                       \
                apptrace.1                      \
                apropos.1                       \
                ar.1                            \
                arch.1                          \
                asa.1                           \
                at.1                            \
                atq.1                           \
                atrm.1                          \
                audioconvert.1                  \
                audioctl.1                      \
                audioplay.1                     \
                audiorecord.1                   \
                audiotest.1                     \
                auths.1                         \
                awk.1                           \
                banner.1                        \
                basename.1                      \
                bc.1                            \
                bdiff.1                         \
                bfs.1                           \
                break.1                         \
                builtin.1                       \
                cal.1                           \
                calendar.1                      \
                cancel.1                        \
                cat.1                           \
                cd.1                            \
                cdrw.1                          \
                checknr.1                       \
                chgrp.1                         \
                chkey.1                         \
                chmod.1                         \
                chown.1                         \
```

```
                ckdate.1                        \
                ckgid.1                         \
                ckint.1                         \
                ckitem.1                        \
                ckkeywd.1                       \
                ckpath.1                        \
                ckrange.1                       \
                ckstr.1                         \
                cksum.1                         \
                cktime.1                        \
                ckuid.1                         \
                ckyorn.1                        \
                clear.1                         \
                cmp.1                           \
                col.1                           \
                comm.1                          \
                command.1                       \
                compress.1                      \
                cp.1                            \
                cpio.1                          \
                cputrack.1                      \
                crle.1                          \
                crontab.1                       \
                crypt.1                         \
                csh.1                           \
                csplit.1                        \
                ctags.1                         \
                ctrun.1                         \
                ctstat.1                        \
                ctwatch.1                       \
                cut.1                           \
                date.1                          \
                dc.1                            \
                deallocate.1                    \
                deroff.1                        \
                dhcpinfo.1                      \
                diff.1                          \
                diff3.1                         \
                diffmk.1                        \
                digest.1                        \
                dircmp.1                        \
                dis.1                           \
                disown.1                        \
                dispgid.1                       \
                dispuid.1                       \
                dos2unix.1                      \
                download.1                      \
                dpost.1                         \
                du.1                            \
                dump.1                          \
                dumpcs.1                        \
                echo.1                          \
                ed.1                            \
                egrep.1                         \
                eject.1                         \
                elfdump.1                       \
                elfedit.1                       \
                elfsign.1                       \
                elfwrap.1                       \
                enable.1                        \
                encrypt.1                       \
                enhance.1                       \
                env.1                           \
                eqn.1                           \
                exec.1                          \
                exit.1                          \
                expand.1                        \
                expr.1                          \
                exstr.1                         \
                factor.1                        \
                fdformat.1                      \
                fgrep.1                         \
                file.1                          \
                filebench.1                     \
                filesync.1                      \
                find.1                          \
                finger.1                        \
                fmt.1                           \
                fmtmsg.1                        \
                fold.1                          \
                ftp.1                           \
                ftpcount.1                      \
                ftpwho.1                        \
                gcore.1                         \
                gencat.1                        \
                genmsg.1                        \
                getconf.1                       \
                getfacl.1                       \
                getlabel.1                      \
                getopt.1                        \
                getoptcvt.1                     \
                getopts.1                       \
                gettext.1                       \
                gettxt.1                        \
                getzonepath.1                   \
                glob.1                          \
                gprof.1                         \
                grep.1                          \
                groups.1                        \
                hash.1                          \
                head.1                          \
                history.1                       \
                hostid.1                        \
                hostname.1                      \
                iconv.1                         \
                indxbib.1                       \
                Intro.1                         \
                ipcrm.1                         \
                ipcs.1                          \
                isainfo.1                       \
                isalist.1                       \
                jobs.1                          \
                join.1                          \
                kbd.1                           \
                kdestroy.1                      \
                keylogin.1                      \
                keylogout.1                     \
                kill.1                          \
                kinit.1                         \
                klist.1                         \
                kmdb.1                          \
                kmfcfg.1                        \
                kpasswd.1                       \
                krb5-config.1                   \
                ksh93.1                         \
                ktutil.1                        \
                lari.1                          \
                last.1                          \
                lastcomm.1                      \
                ld.1                            \
                ldap.1                          \
                ldapdelete.1                    \
```

```
                ldaplist.1                      \
                ldapmodify.1                    \
                ldapmodrdn.1                    \
                ldapsearch.1                    \
                ldd.1                           \
                ld.so.1.1                       \
                let.1                           \
                lex.1                           \
                lgrpinfo.1                      \
                limit.1                         \
                line.1                          \
                list_devices.1                  \
                listusers.1                     \
                ln.1                            \
                loadkeys.1                      \
                locale.1                        \
                localedef.1                     \
                logger.1                        \
                login.1                         \
                logname.1                       \
                logout.1                        \
                look.1                          \
                lookbib.1                       \
                lorder.1                        \
                lp.1                            \
                lpstat.1                        \
                ls.1                            \
                m4.1                            \
                mac.1                           \
                mach.1                          \
                madv.so.1.1                     \
                mail.1                          \
                mailcompat.1                    \
                mailq.1                         \
                mailstats.1                     \
                mailx.1                         \
                makekey.1                       \
                man.1                           \
                mconnect.1                      \
                mcs.1                           \
                mdb.1                           \
                mesg.1                          \
                mkdir.1                         \
                mkmsgs.1                        \
                mktemp.1                        \
                moe.1                           \
                more.1                          \
                mpss.so.1.1                     \
                msgcc.1                         \
                msgcpp.1                        \
                msgcvt.1                        \
                msgfmt.1                        \
                msggen.1                        \
                msgget.1                        \
                mt.1                            \
                mv.1                            \
                nawk.1                          \
                nc.1                            \
                nca.1                           \
                ncab2clf.1                      \
                ncakmod.1                       \
                newform.1                       \
                newgrp.1                        \
                news.1                          \
                newtask.1                       \
                nice.1                          \
                nl.1                            \
                nm.1                            \
                nohup.1                         \
                nroff.1                         \
                od.1                            \
                on.1                            \
                optisa.1                        \
                pack.1                          \
                pagesize.1                      \
                pargs.1                         \
                passwd.1                        \
                paste.1                         \
                pathchk.1                       \
                pax.1                           \
                pfexec.1                        \
                pg.1                            \
                pgrep.1                         \
                pkginfo.1                       \
                pkgmk.1                         \
                pkgparam.1                      \
                pkgproto.1                      \
                pkgtrans.1                      \
                pktool.1                        \
                plabel.1                        \
                plgrp.1                         \
                plimit.1                        \
                pmadvise.1                      \
                pmap.1                          \
                postio.1                        \
                postprint.1                     \
                postreverse.1                   \
                ppgsz.1                         \
                ppriv.1                         \
                pr.1                            \
                praliases.1                     \
                prctl.1                         \
                preap.1                         \
                prex.1                          \
                print.1                         \
                printf.1                        \
                priocntl.1                      \
                proc.1                          \
                prof.1                          \
                profiles.1                      \
                projects.1                      \
                ps.1                            \
                ptree.1                         \
                pvs.1                           \
                pwd.1                           \
                ranlib.1                        \
                rcapstat.1                      \
                rcp.1                           \
                rdist.1                         \
                read.1                          \
                readonly.1                      \
                refer.1                         \
                regcmp.1                        \
                renice.1                        \
                rev.1                           \
                rlogin.1                        \
                rm.1                            \
                rmformat.1                      \
                rmmount.1                       \
                roffbib.1                       \
                roles.1                         \
                rpcgen.1                        \
```

```
                rsh.1                           \
                runat.1                         \
                rup.1                           \
                ruptime.1                       \
                rusers.1                        \
                rwho.1                          \
                sar.1                           \
                scp.1                           \
                script.1                        \
                sdiff.1                         \
                sed.1                           \
                set.1                           \
                setfacl.1                       \
                setlabel.1                      \
                setpgrp.1                       \
                sftp.1                          \
                shcomp.1                        \
                shell_builtins.1                \
                shift.1                         \
                size.1                          \
                sleep.1                         \
                smbutil.1                       \
                soelim.1                        \
                sort.1                          \
                sortbib.1                       \
                sotruss.1                       \
                spell.1                         \
                split.1                         \
                srchtxt.1                       \
                ssh.1                           \
                ssh-add.1                       \
                ssh-agent.1                     \
                ssh-http-proxy-connect.1        \
                ssh-keygen.1                    \
                ssh-keyscan.1                   \
                ssh-socks5-proxy-connect.1      \
                strchg.1                        \
                strings.1                       \
                strip.1                         \
                stty.1                          \
                sum.1                           \
                suspend.1                       \
                svcprop.1                       \
                svcs.1                          \
                symorder.1                      \
                sys-suspend.1                   \
                tabs.1                          \
                tail.1                          \
                talk.1                          \
                tar.1                           \
                tbl.1                           \
                tcopy.1                         \
                tee.1                           \
                telnet.1                        \
                test.1                          \
                tftp.1                          \
                time.1                          \
                times.1                         \
                timex.1                         \
                tip.1                           \
                tnfdump.1                       \
                tnfxtract.1                     \
                touch.1                         \
                tput.1                          \
                tr.1                            \
                trap.1                          \
                troff.1                         \
                true.1                          \
                truss.1                         \
                tsort.1                         \
                tty.1                           \
                type.1                          \
                typeset.1                       \
                ul.1                            \
                umask.1                         \
                uname.1                         \
                unifdef.1                       \
                uniq.1                          \
                units.1                         \
                unix2dos.1                      \
                uptime.1                        \
                vacation.1                      \
                vgrind.1                        \
                volcheck.1                      \
                volrmmount.1                    \
                w.1                             \
                wait.1                          \
                wc.1                            \
                whatis.1                        \
                which.1                         \
                who.1                           \
                whocalls.1                      \
                whois.1                         \
                write.1                         \
                xargs.1                         \
                xgettext.1                      \
                xstr.1                          \
                yacc.1                          \
                yes.1                           \
                ypcat.1                         \
                ypmatch.1                       \
                yppasswd.1                      \
                ypwhich.1                       \
                zlogin.1                        \
                zonename.1

MANLINKS=       batch.1         \
                bg.1            \
                case.1          \
                chdir.1         \
                checkeq.1       \
                continue.1      \
                decrypt.1       \
                dirname.1       \
                dirs.1          \
                disable.1       \
                dumpkeys.1      \
                edit.1          \
                errange.1       \
                errdate.1       \
                errgid.1        \
                errint.1        \
                erritem.1       \
                errpath.1       \
                errstr.1        \
                errtime.1       \
                erruid.1        \
                erryorn.1       \
                eval.1          \
                export.1        \
                false.1         \
                fc.1            \
```

```
                fg.1            \
                for.1           \
                foreach.1       \
                function.1      \
                goto.1          \
                hashcheck.1     \
                hashmake.1      \
                hashstat.1      \
                helpdate.1      \
                helpgid.1       \
                helpint.1       \
                helpitem.1      \
                helppath.1      \
                helprange.1     \
                helpstr.1       \
                helptime.1      \
                helpuid.1       \
                helpyorn.1      \
                hist.1          \
                if.1            \
                intro.1         \
                jsh.1           \
                ksh.1           \
                ldapadd.1       \
                neqn.1          \
                notify.1        \
                onintr.1        \
                page.1          \
                pcat.1          \
                pcred.1         \
                pfcsh.1         \
                pfiles.1        \
                pfksh.1         \
                pflags.1        \
                pfsh.1          \
                pkill.1         \
                pldd.1          \
                popd.1          \
                prun.1          \
                psig.1          \
                pstack.1        \
                pstop.1         \
                ptime.1         \
                pushd.1         \
                pwait.1         \
                pwdx.1          \
                red.1           \
                rehash.1        \
                remote_shell.1  \
                remsh.1         \
                repeat.1        \
                return.1        \
                rksh.1          \
                rksh93.1        \
                rmail.1         \
                rmdir.1         \
                rmumount.1      \
                select.1        \
                setenv.1        \
                settime.1       \
                sh.1            \
                snca.1          \
                source.1        \
                spellin.1       \
                stop.1          \
                strconf.1       \
                switch.1        \
                ulimit.1        \
                unalias.1       \
                uncompress.1    \
                unexpand.1      \
                unhash.1        \
                unlimit.1       \
                unpack.1        \
                unset.1         \
                unsetenv.1      \
                until.1         \
                valdate.1       \
                valgid.1        \
                valint.1        \
                valpath.1       \
                valrange.1      \
                valstr.1        \
                valtime.1       \
                valuid.1        \
                valyorn.1       \
                vax.1           \
                vedit.1         \
                whence.1        \
                while.1         \
                zcat.1

intro.1         := LINKSRC = Intro.1

unalias.1       := LINKSRC = alias.1

batch.1         := LINKSRC = at.1

dirname.1       := LINKSRC = basename.1

continue.1      := LINKSRC = break.1

chdir.1         := LINKSRC = cd.1
dirs.1          := LINKSRC = cd.1
popd.1          := LINKSRC = cd.1
pushd.1         := LINKSRC = cd.1

errdate.1       := LINKSRC = ckdate.1
helpdate.1      := LINKSRC = ckdate.1
valdate.1       := LINKSRC = ckdate.1

errgid.1        := LINKSRC = ckgid.1
helpgid.1       := LINKSRC = ckgid.1
valgid.1        := LINKSRC = ckgid.1

errint.1        := LINKSRC = ckint.1
helpint.1       := LINKSRC = ckint.1
valint.1        := LINKSRC = ckint.1

erritem.1       := LINKSRC = ckitem.1
helpitem.1      := LINKSRC = ckitem.1

errpath.1       := LINKSRC = ckpath.1
helppath.1      := LINKSRC = ckpath.1
valpath.1       := LINKSRC = ckpath.1

errange.1       := LINKSRC = ckrange.1
helprange.1     := LINKSRC = ckrange.1
valrange.1      := LINKSRC = ckrange.1

errstr.1        := LINKSRC = ckstr.1
helpstr.1       := LINKSRC = ckstr.1
```

```
valstr.1        := LINKSRC = ckstr.1

errtime.1       := LINKSRC = cktime.1
helptime.1      := LINKSRC = cktime.1
valtime.1       := LINKSRC = cktime.1

erruid.1        := LINKSRC = ckuid.1
helpuid.1       := LINKSRC = ckuid.1
valuid.1        := LINKSRC = ckuid.1

erryorn.1       := LINKSRC = ckyorn.1
helpyorn.1      := LINKSRC = ckyorn.1
valyorn.1       := LINKSRC = ckyorn.1

uncompress.1    := LINKSRC = compress.1
zcat.1          := LINKSRC = compress.1

red.1           := LINKSRC = ed.1

disable.1       := LINKSRC = enable.1

decrypt.1       := LINKSRC = encrypt.1

checkeq.1       := LINKSRC = eqn.1
neqn.1          := LINKSRC = eqn.1

eval.1          := LINKSRC = exec.1
source.1        := LINKSRC = exec.1

goto.1          := LINKSRC = exit.1
return.1        := LINKSRC = exit.1

unexpand.1      := LINKSRC = expand.1

hashstat.1      := LINKSRC = hash.1
rehash.1        := LINKSRC = hash.1
unhash.1        := LINKSRC = hash.1

fc.1            := LINKSRC = history.1
hist.1          := LINKSRC = history.1

bg.1            := LINKSRC = jobs.1
fg.1            := LINKSRC = jobs.1
notify.1        := LINKSRC = jobs.1
stop.1          := LINKSRC = jobs.1

jsh.1           := LINKSRC = ksh93.1
ksh.1           := LINKSRC = ksh93.1
rksh.1          := LINKSRC = ksh93.1
rksh93.1        := LINKSRC = ksh93.1
sh.1            := LINKSRC = ksh93.1

ldapadd.1       := LINKSRC = ldapmodify.1

ulimit.1        := LINKSRC = limit.1
unlimit.1       := LINKSRC = limit.1

dumpkeys.1      := LINKSRC = loadkeys.1

i286.1          := LINKSRC = machid.1
i386.1          := LINKSRC = machid.1
i486.1          := LINKSRC = machid.1
i860.1          := LINKSRC = machid.1
iAPX286.1       := LINKSRC = machid.1
pdp11.1         := LINKSRC = machid.1
sparc.1         := LINKSRC = machid.1
```

```
sun.1           := LINKSRC = machid.1
u370.1          := LINKSRC = machid.1
u3b.1           := LINKSRC = machid.1
u3b15.1         := LINKSRC = machid.1
u3b2.1          := LINKSRC = machid.1
u3b5.1          := LINKSRC = machid.1
vax.1           := LINKSRC = machid.1

rmail.1         := LINKSRC = mail.1

page.1          := LINKSRC = more.1

snca.1          := LINKSRC = nca.1

pcat.1          := LINKSRC = pack.1
unpack.1        := LINKSRC = pack.1

pfcsh.1         := LINKSRC = pfexec.1
pfksh.1         := LINKSRC = pfexec.1
pfsh.1          := LINKSRC = pfexec.1

pkill.1         := LINKSRC = pgrep.1

pcred.1         := LINKSRC = proc.1
pfiles.1        := LINKSRC = proc.1
pflags.1        := LINKSRC = proc.1
pldd.1          := LINKSRC = proc.1
prun.1          := LINKSRC = proc.1
psig.1          := LINKSRC = proc.1
pstack.1        := LINKSRC = proc.1
pstop.1         := LINKSRC = proc.1
ptime.1         := LINKSRC = proc.1
pwait.1         := LINKSRC = proc.1
pwdx.1          := LINKSRC = proc.1

rmdir.1         := LINKSRC = rm.1

rmumount.1      := LINKSRC = rmmount.1

remote_shell.1  := LINKSRC = rsh.1
remsh.1         := LINKSRC = rsh.1

export.1        := LINKSRC = set.1
setenv.1        := LINKSRC = set.1
unset.1         := LINKSRC = set.1
unsetenv.1      := LINKSRC = set.1

case.1          := LINKSRC = shell_builtins.1
for.1           := LINKSRC = shell_builtins.1
foreach.1       := LINKSRC = shell_builtins.1
function.1      := LINKSRC = shell_builtins.1
if.1            := LINKSRC = shell_builtins.1
repeat.1        := LINKSRC = shell_builtins.1
select.1        := LINKSRC = shell_builtins.1
switch.1        := LINKSRC = shell_builtins.1
until.1         := LINKSRC = shell_builtins.1
while.1         := LINKSRC = shell_builtins.1

hashcheck.1     := LINKSRC = spell.1
hashmake.1      := LINKSRC = spell.1
spellin.1       := LINKSRC = spell.1

strconf.1       := LINKSRC = strchg.1

settime.1       := LINKSRC = touch.1
```

```
onintr.1        := LINKSRC = trap.1

false.1         := LINKSRC = true.1

whence.1        := LINKSRC = typeset.1

# Links to usr/has/man

edit.1          := LINKSRC = ../../../has/man/man1has/edit.1has

vedit.1         := LINKSRC = ../../../has/man/man1has/vi.1has

.KEEP_STATE:

include         $(SRC)/man/Makefile.man

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

```
**********************************************************
   13156 Thu Feb 13 22:45:27 2014
new/usr/src/pkg/manifests/SUNWcs.man1.inc
remove machid manual pages, since machid was removed
**********************************************************
#
# This file and its contents are supplied under the terms of the
# Common Development and Distribution License ("CDDL"), version 1.0.
# You may only use this file in accordance with the terms of version
# 1.0 of the CDDL.
#
# A full copy of the text of the CDDL should have accompanied this
# source.  A copy of the CDDL is also available via the Internet
# at http://www.illumos.org/license/CDDL.
#

#
# Copyright 2011, Richard Lowe
# Copyright 2012 OmniTI Computer Consulting, Inc.  All rights reserved.
# Copyright 2012 Nexenta Systems, Inc. All rights reserved.
#

file path=usr/share/man/man1/Intro.1
file path=usr/share/man/man1/alias.1
file path=usr/share/man/man1/allocate.1
file path=usr/share/man/man1/amt.1
file path=usr/share/man/man1/arch.1
file path=usr/share/man/man1/at.1
file path=usr/share/man/man1/atq.1
file path=usr/share/man/man1/atrm.1
file path=usr/share/man/man1/auths.1
file path=usr/share/man/man1/basename.1
file path=usr/share/man/man1/break.1
file path=usr/share/man/man1/builtin.1
file path=usr/share/man/man1/cat.1
file path=usr/share/man/man1/cd.1
file path=usr/share/man/man1/chgrp.1
file path=usr/share/man/man1/chmod.1
file path=usr/share/man/man1/chown.1
file path=usr/share/man/man1/ckdate.1
file path=usr/share/man/man1/ckgid.1
file path=usr/share/man/man1/ckint.1
file path=usr/share/man/man1/ckitem.1
file path=usr/share/man/man1/ckkeywd.1
file path=usr/share/man/man1/ckpath.1
file path=usr/share/man/man1/ckrange.1
file path=usr/share/man/man1/ckstr.1
file path=usr/share/man/man1/cksum.1
file path=usr/share/man/man1/cktime.1
file path=usr/share/man/man1/ckuid.1
file path=usr/share/man/man1/ckyorn.1
file path=usr/share/man/man1/clear.1
file path=usr/share/man/man1/cmp.1
file path=usr/share/man/man1/comm.1
file path=usr/share/man/man1/command.1
file path=usr/share/man/man1/cp.1
file path=usr/share/man/man1/cpio.1
file path=usr/share/man/man1/crle.1
file path=usr/share/man/man1/crontab.1
file path=usr/share/man/man1/crypt.1
file path=usr/share/man/man1/csh.1
file path=usr/share/man/man1/ctrun.1
file path=usr/share/man/man1/ctstat.1
file path=usr/share/man/man1/ctwatch.1
file path=usr/share/man/man1/cut.1
file path=usr/share/man/man1/date.1
```

```
file path=usr/share/man/man1/deallocate.1
file path=usr/share/man/man1/dhcpinfo.1
file path=usr/share/man/man1/digest.1
file path=usr/share/man/man1/disown.1
file path=usr/share/man/man1/dispgid.1
file path=usr/share/man/man1/dispuid.1
file path=usr/share/man/man1/du.1
file path=usr/share/man/man1/dumpcs.1
file path=usr/share/man/man1/echo.1
file path=usr/share/man/man1/ed.1
file path=usr/share/man/man1/egrep.1
file path=usr/share/man/man1/eject.1
file path=usr/share/man/man1/encrypt.1
file path=usr/share/man/man1/env.1
file path=usr/share/man/man1/exec.1
file path=usr/share/man/man1/exit.1
file path=usr/share/man/man1/expr.1
file path=usr/share/man/man1/fdformat.1
file path=usr/share/man/man1/fgrep.1
file path=usr/share/man/man1/file.1
file path=usr/share/man/man1/find.1
file path=usr/share/man/man1/fmt.1
file path=usr/share/man/man1/fmtmsg.1
file path=usr/share/man/man1/fold.1
file path=usr/share/man/man1/getconf.1
file path=usr/share/man/man1/getfacl.1
file path=usr/share/man/man1/getopt.1
file path=usr/share/man/man1/getoptcvt.1
file path=usr/share/man/man1/getopts.1
file path=usr/share/man/man1/gettext.1
file path=usr/share/man/man1/glob.1
file path=usr/share/man/man1/grep.1
file path=usr/share/man/man1/groups.1
file path=usr/share/man/man1/hash.1
file path=usr/share/man/man1/head.1
file path=usr/share/man/man1/history.1
file path=usr/share/man/man1/hostid.1
file path=usr/share/man/man1/hostname.1
file path=usr/share/man/man1/iconv.1
file path=usr/share/man/man1/isainfo.1
file path=usr/share/man/man1/isalist.1
file path=usr/share/man/man1/jobs.1
file path=usr/share/man/man1/join.1
file path=usr/share/man/man1/kbd.1
file path=usr/share/man/man1/keylogin.1
file path=usr/share/man/man1/keylogout.1
file path=usr/share/man/man1/kill.1
file path=usr/share/man/man1/kmfcfg.1
file path=usr/share/man/man1/ksh93.1
file path=usr/share/man/man1/ld.so.1.1
file path=usr/share/man/man1/let.1
file path=usr/share/man/man1/limit.1
file path=usr/share/man/man1/line.1
file path=usr/share/man/man1/list_devices.1
file path=usr/share/man/man1/listusers.1
file path=usr/share/man/man1/ln.1
file path=usr/share/man/man1/loadkeys.1
file path=usr/share/man/man1/localedef.1
file path=usr/share/man/man1/logger.1
file path=usr/share/man/man1/login.1
file path=usr/share/man/man1/logname.1
file path=usr/share/man/man1/logout.1
file path=usr/share/man/man1/ls.1
file path=usr/share/man/man1/m4.1
file path=usr/share/man/man1/mac.1
file path=usr/share/man/man1/mach.1
file path=usr/share/man/man1/mail.1
file path=usr/share/man/man1/mailx.1
file path=usr/share/man/man1/makekey.1
file path=usr/share/man/man1/mesg.1
file path=usr/share/man/man1/mkdir.1
file path=usr/share/man/man1/mktemp.1
file path=usr/share/man/man1/moe.1
file path=usr/share/man/man1/more.1
file path=usr/share/man/man1/mt.1
file path=usr/share/man/man1/mv.1
file path=usr/share/man/man1/newgrp.1
file path=usr/share/man/man1/newtask.1
file path=usr/share/man/man1/nice.1
file path=usr/share/man/man1/nohup.1
file path=usr/share/man/man1/optisa.1
file path=usr/share/man/man1/pagesize.1
file path=usr/share/man/man1/passwd.1
file path=usr/share/man/man1/paste.1
file path=usr/share/man/man1/pathchk.1
file path=usr/share/man/man1/pax.1
file path=usr/share/man/man1/pfexec.1
file path=usr/share/man/man1/pg.1
file path=usr/share/man/man1/pgrep.1
file path=usr/share/man/man1/pktool.1
file path=usr/share/man/man1/pr.1
file path=usr/share/man/man1/prctl.1
file path=usr/share/man/man1/print.1
file path=usr/share/man/man1/priocntl.1
file path=usr/share/man/man1/proc.1
file path=usr/share/man/man1/profiles.1
file path=usr/share/man/man1/projects.1
file path=usr/share/man/man1/ps.1
file path=usr/share/man/man1/pwd.1
file path=usr/share/man/man1/read.1
file path=usr/share/man/man1/readonly.1
file path=usr/share/man/man1/renice.1
file path=usr/share/man/man1/rev.1
file path=usr/share/man/man1/rm.1
file path=usr/share/man/man1/roles.1
file path=usr/share/man/man1/runat.1
file path=usr/share/man/man1/script.1
file path=usr/share/man/man1/sed.1
file path=usr/share/man/man1/set.1
file path=usr/share/man/man1/setfacl.1
file path=usr/share/man/man1/setpgrp.1
file path=usr/share/man/man1/shcomp.1
file path=usr/share/man/man1/shell_builtins.1
file path=usr/share/man/man1/shift.1
file path=usr/share/man/man1/sleep.1
file path=usr/share/man/man1/strchg.1
file path=usr/share/man/man1/stty.1
file path=usr/share/man/man1/sum.1
file path=usr/share/man/man1/suspend.1
file path=usr/share/man/man1/svcprop.1
file path=usr/share/man/man1/svcs.1
file path=usr/share/man/man1/tabs.1
file path=usr/share/man/man1/tail.1
file path=usr/share/man/man1/tar.1
file path=usr/share/man/man1/tee.1
file path=usr/share/man/man1/test.1
file path=usr/share/man/man1/time.1
file path=usr/share/man/man1/times.1
file path=usr/share/man/man1/tip.1
file path=usr/share/man/man1/touch.1
file path=usr/share/man/man1/tput.1
file path=usr/share/man/man1/tr.1
```

```
file path=usr/share/man/man1/trap.1
file path=usr/share/man/man1/true.1
file path=usr/share/man/man1/tty.1
file path=usr/share/man/man1/type.1
file path=usr/share/man/man1/typeset.1
file path=usr/share/man/man1/umask.1
file path=usr/share/man/man1/uname.1
file path=usr/share/man/man1/uniq.1
file path=usr/share/man/man1/uptime.1
file path=usr/share/man/man1/w.1
file path=usr/share/man/man1/wait.1
file path=usr/share/man/man1/wc.1
file path=usr/share/man/man1/which.1
file path=usr/share/man/man1/who.1
file path=usr/share/man/man1/write.1
file path=usr/share/man/man1/xargs.1
file path=usr/share/man/man1/xstr.1
file path=usr/share/man/man1/zonename.1
link path=usr/share/man/man1/batch.1 target=at.1
link path=usr/share/man/man1/bg.1 target=jobs.1
link path=usr/share/man/man1/case.1 target=shell_builtins.1
link path=usr/share/man/man1/chdir.1 target=cd.1
link path=usr/share/man/man1/continue.1 target=break.1
link path=usr/share/man/man1/decrypt.1 target=encrypt.1
link path=usr/share/man/man1/dirname.1 target=basename.1
link path=usr/share/man/man1/dirs.1 target=cd.1
link path=usr/share/man/man1/dumpkeys.1 target=loadkeys.1
link path=usr/share/man/man1/edit.1 target=../../../has/man/man1has/edit.1has
link path=usr/share/man/man1/errange.1 target=ckrange.1
link path=usr/share/man/man1/errdate.1 target=ckdate.1
link path=usr/share/man/man1/errgid.1 target=ckgid.1
link path=usr/share/man/man1/errint.1 target=ckint.1
link path=usr/share/man/man1/erritem.1 target=ckitem.1
link path=usr/share/man/man1/errpath.1 target=ckpath.1
link path=usr/share/man/man1/errstr.1 target=ckstr.1
link path=usr/share/man/man1/errtime.1 target=cktime.1
link path=usr/share/man/man1/erruid.1 target=ckuid.1
link path=usr/share/man/man1/erryorn.1 target=ckyorn.1
link path=usr/share/man/man1/eval.1 target=exec.1
link path=usr/share/man/man1/export.1 target=set.1
link path=usr/share/man/man1/false.1 target=true.1
link path=usr/share/man/man1/fc.1 target=history.1
link path=usr/share/man/man1/fg.1 target=jobs.1
link path=usr/share/man/man1/for.1 target=shell_builtins.1
link path=usr/share/man/man1/foreach.1 target=shell_builtins.1
link path=usr/share/man/man1/function.1 target=shell_builtins.1
link path=usr/share/man/man1/goto.1 target=exit.1
link path=usr/share/man/man1/hashstat.1 target=hash.1
link path=usr/share/man/man1/helpdate.1 target=ckdate.1
link path=usr/share/man/man1/helpgid.1 target=ckgid.1
link path=usr/share/man/man1/helpint.1 target=ckint.1
link path=usr/share/man/man1/helpitem.1 target=ckitem.1
link path=usr/share/man/man1/helppath.1 target=ckpath.1
link path=usr/share/man/man1/helprange.1 target=ckrange.1
link path=usr/share/man/man1/helpstr.1 target=ckstr.1
link path=usr/share/man/man1/helptime.1 target=cktime.1
link path=usr/share/man/man1/helpuid.1 target=ckuid.1
link path=usr/share/man/man1/helpyorn.1 target=ckyorn.1
link path=usr/share/man/man1/hist.1 target=history.1
link path=usr/share/man/man1/i286.1 target=machid.1
link path=usr/share/man/man1/i386.1 target=machid.1
link path=usr/share/man/man1/i486.1 target=machid.1
link path=usr/share/man/man1/i860.1 target=machid.1
link path=usr/share/man/man1/iAPX286.1 target=machid.1
link path=usr/share/man/man1/if.1 target=shell_builtins.1
link path=usr/share/man/man1/intro.1 target=Intro.1
link path=usr/share/man/man1/jsh.1 target=ksh93.1
link path=usr/share/man/man1/ksh.1 target=ksh93.1
link path=usr/share/man/man1/notify.1 target=jobs.1
link path=usr/share/man/man1/onintr.1 target=trap.1
link path=usr/share/man/man1/page.1 target=more.1
link path=usr/share/man/man1/pcred.1 target=proc.1
link path=usr/share/man/man1/pdp11.1 target=machid.1
link path=usr/share/man/man1/pfcsh.1 target=pfexec.1
link path=usr/share/man/man1/pfiles.1 target=proc.1
link path=usr/share/man/man1/pfksh.1 target=pfexec.1
link path=usr/share/man/man1/pflags.1 target=proc.1
link path=usr/share/man/man1/pfsh.1 target=pfexec.1
link path=usr/share/man/man1/pkill.1 target=pgrep.1
link path=usr/share/man/man1/pldd.1 target=proc.1
link path=usr/share/man/man1/popd.1 target=cd.1
link path=usr/share/man/man1/prun.1 target=proc.1
link path=usr/share/man/man1/psig.1 target=proc.1
link path=usr/share/man/man1/pstack.1 target=proc.1
link path=usr/share/man/man1/pstop.1 target=proc.1
link path=usr/share/man/man1/ptime.1 target=proc.1
link path=usr/share/man/man1/pushd.1 target=cd.1
link path=usr/share/man/man1/pwait.1 target=proc.1
link path=usr/share/man/man1/pwdx.1 target=proc.1
link path=usr/share/man/man1/red.1 target=ed.1
link path=usr/share/man/man1/rehash.1 target=hash.1
link path=usr/share/man/man1/repeat.1 target=shell_builtins.1
link path=usr/share/man/man1/return.1 target=exit.1
link path=usr/share/man/man1/rksh.1 target=ksh93.1
link path=usr/share/man/man1/rksh93.1 target=ksh93.1
link path=usr/share/man/man1/rmail.1 target=mail.1
link path=usr/share/man/man1/rmdir.1 target=rm.1
link path=usr/share/man/man1/select.1 target=shell_builtins.1
link path=usr/share/man/man1/setenv.1 target=set.1
link path=usr/share/man/man1/settime.1 target=touch.1
link path=usr/share/man/man1/sh.1 target=ksh93.1
link path=usr/share/man/man1/source.1 target=exec.1
link path=usr/share/man/man1/sparc.1 target=machid.1
link path=usr/share/man/man1/stop.1 target=jobs.1
link path=usr/share/man/man1/strconf.1 target=strchg.1
link path=usr/share/man/man1/sun.1 target=machid.1
link path=usr/share/man/man1/switch.1 target=shell_builtins.1
link path=usr/share/man/man1/u370.1 target=machid.1
link path=usr/share/man/man1/u3b.1 target=machid.1
link path=usr/share/man/man1/u3b15.1 target=machid.1
link path=usr/share/man/man1/u3b2.1 target=machid.1
link path=usr/share/man/man1/u3b5.1 target=machid.1
link path=usr/share/man/man1/ulimit.1 target=limit.1
link path=usr/share/man/man1/unalias.1 target=alias.1
link path=usr/share/man/man1/unhash.1 target=hash.1
link path=usr/share/man/man1/unlimit.1 target=limit.1
link path=usr/share/man/man1/unset.1 target=set.1
link path=usr/share/man/man1/unsetenv.1 target=set.1
link path=usr/share/man/man1/until.1 target=shell_builtins.1
link path=usr/share/man/man1/valdate.1 target=ckdate.1
link path=usr/share/man/man1/valgid.1 target=ckgid.1
link path=usr/share/man/man1/valint.1 target=ckint.1
link path=usr/share/man/man1/valpath.1 target=ckpath.1
link path=usr/share/man/man1/valrange.1 target=ckrange.1
link path=usr/share/man/man1/valstr.1 target=ckstr.1
link path=usr/share/man/man1/valtime.1 target=cktime.1
link path=usr/share/man/man1/valuid.1 target=ckuid.1
link path=usr/share/man/man1/valyorn.1 target=ckyorn.1
link path=usr/share/man/man1/vax.1 target=machid.1
link path=usr/share/man/man1/vedit.1 target=../../../has/man/man1has/vi.1has
link path=usr/share/man/man1/whence.1 target=typeset.1
link path=usr/share/man/man1/while.1 target=shell_builtins.1
```

```
**********************************************************
   26469 Thu Feb 13 22:45:27 2014
new/usr/src/uts/common/os/grow.c
os/grow: define 'p' under the same ifdef as it's consumed
**********************************************************
______unchanged_portion_omitted_

/*
 * Find address for user to map.
 * If MAP_FIXED is not specified, we can pick any address we want, but we will
 * first try the value in *addrp if it is non-NULL.  Thus this is implementing
 * a way to try and get a preferred address.
 */
int
choose_addr(struct as *as, caddr_t *addrp, size_t len, offset_t off,
    int vacalign, uint_t flags)
{
#if defined(__amd64)
#endif /* ! codereview */
        proc_t *p = curproc;
#endif
#endif /* ! codereview */
        caddr_t basep = (caddr_t)(uintptr_t)((uintptr_t)*addrp & PAGEMASK);
        size_t lenp;

        ASSERT(AS_ISCLAIMGAP(as));      /* searches should be serialized */

        /*
         * If we have been provided a hint, we should still expand the lenp
         * to be the rest of the address space.  This will allow us to
         * treat the hint as a strong desire to be "nearby" the provided
         * address.  If we can't satisfy the hint, as_gap() will walk forward.
         */
        if (flags & _MAP_LOW32)
                lenp = (caddr_t)USERLIMIT32 - basep;
#if defined(__amd64)
        else if (p->p_model == DATAMODEL_NATIVE)
                lenp = p->p_usrstack - basep -
                    ((p->p_stk_ctl + PAGEOFFSET) & PAGEMASK);
#endif
        else
                lenp = as->a_userlimit - basep;

        if (flags & MAP_FIXED) {
                (void) as_unmap(as, *addrp, len);
                return (0);
        } else if (basep != NULL && ((flags & MAP_ALIGN) == 0) &&
            !as_gap(as, len, &basep, &lenp, 0, *addrp)) {
                /* User supplied address was available */
                *addrp = basep;
        } else {
                /*
                 * No user supplied address or the address supplied was not
                 * available.
                 */
                map_addr(addrp, len, off, vacalign, flags);
        }
        if (*addrp == NULL)
                return (ENOMEM);
        return (0);
}


/*
 * Used for MAP_ANON - fast way to get anonymous pages
 */
```

```
static int
zmap(struct as *as, caddr_t *addrp, size_t len, uint_t uprot, int flags,
    offset_t pos)
{
        struct segvn_crargs vn_a;
        int error;

        if (((PROT_ALL & uprot) != uprot))
                return (EACCES);

        if ((flags & MAP_FIXED) != 0) {
                caddr_t userlimit;

                /*
                 * Use the user address.  First verify that
                 * the address to be used is page aligned.
                 * Then make some simple bounds checks.
                 */
                if (((uintptr_t)*addrp & PAGEOFFSET) != 0)
                        return (EINVAL);

                userlimit = flags & _MAP_LOW32 ?
                    (caddr_t)USERLIMIT32 : as->a_userlimit;
                switch (valid_usr_range(*addrp, len, uprot, as, userlimit)) {
                case RANGE_OKAY:
                        break;
                case RANGE_BADPROT:
                        return (ENOTSUP);
                case RANGE_BADADDR:
                default:
                        return (ENOMEM);
                }
        }
        /*
         * No need to worry about vac alignment for anonymous
         * pages since this is a "clone" object that doesn't
         * yet exist.
         */
        error = choose_addr(as, addrp, len, pos, ADDR_NOVACALIGN, flags);
        if (error != 0) {
                return (error);
        }

        /*
         * Use the seg_vn segment driver; passing in the NULL amp
         * gives the desired "cloning" effect.
         */
        vn_a.vp = NULL;
        vn_a.offset = 0;
        vn_a.type = flags & MAP_TYPE;
        vn_a.prot = uprot;
        vn_a.maxprot = PROT_ALL;
        vn_a.flags = flags & ~MAP_TYPE;
        vn_a.cred = CRED();
        vn_a.amp = NULL;
        vn_a.szc = 0;
        vn_a.lgrp_mem_policy_flags = 0;

        return (as_map(as, *addrp, len, segvn_create, &vn_a));
}

static int
smmap_common(caddr_t *addrp, size_t len,
    int prot, int flags, struct file *fp, offset_t pos)
{
        struct vnode *vp;
        struct as *as = curproc->p_as;
        uint_t uprot, maxprot, type;
        int error;
        int in_crit = 0;

        if ((flags & ~(MAP_SHARED | MAP_PRIVATE | MAP_FIXED | _MAP_NEW |
            _MAP_LOW32 | MAP_NORESERVE | MAP_ANON | MAP_ALIGN |
            MAP_TEXT | MAP_INITDATA)) != 0) {
                /* | MAP_RENAME */      /* not implemented, let user know */
                return (EINVAL);
        }

        if ((flags & MAP_TEXT) && !(prot & PROT_EXEC)) {
                return (EINVAL);
        }

        if ((flags & (MAP_TEXT | MAP_INITDATA)) == (MAP_TEXT | MAP_INITDATA)) {
                return (EINVAL);
        }

#if defined(__sparc)
        /*
         * See if this is an "old mmap call".  If so, remember this
         * fact and convert the flags value given to mmap to indicate
         * the specified address in the system call must be used.
         * _MAP_NEW is turned set by all new uses of mmap.
         */
        if ((flags & _MAP_NEW) == 0)
                flags |= MAP_FIXED;
#endif
        flags &= ~_MAP_NEW;

        type = flags & MAP_TYPE;
        if (type != MAP_PRIVATE && type != MAP_SHARED)
                return (EINVAL);


        if (flags & MAP_ALIGN) {

                if (flags & MAP_FIXED)
                        return (EINVAL);

                /* alignment needs to be a power of 2 >= page size */
                if (((uintptr_t)*addrp < PAGESIZE && (uintptr_t)*addrp != 0) ||
                    !ISP2((uintptr_t)*addrp))
                        return (EINVAL);
        }
        /*
         * Check for bad lengths and file position.
         * We let the VOP_MAP routine check for negative lengths
         * since on some vnode types this might be appropriate.
         */
        if (len == 0 || (pos & (u_offset_t)PAGEOFFSET) != 0)
                return (EINVAL);

        maxprot = PROT_ALL;             /* start out allowing all accesses */
        uprot = prot | PROT_USER;

        if (fp == NULL) {
                ASSERT(flags & MAP_ANON);
                /* discard lwpchan mappings, like munmap() */
                if ((flags & MAP_FIXED) && curproc->p_lcp != NULL)
                        lwpchan_delete_mapping(curproc, *addrp, *addrp + len);
                as_rangelock(as);
                error = zmap(as, addrp, len, uprot, flags, pos);
                as_rangeunlock(as);
```

```
                /*
                 * Tell machine specific code that lwp has mapped shared memory
                 */
                if (error == 0 && (flags & MAP_SHARED)) {
                        /* EMPTY */
                        LWP_MMODEL_SHARED_AS(*addrp, len);
                }
                return (error);
        } else if ((flags & MAP_ANON) != 0)
                return (EINVAL);

        vp = fp->f_vnode;

        /* Can’t execute code from "noexec" mounted filesystem. */
        if ((vp->v_vfsp->vfs_flag & VFS_NOEXEC) != 0)
                maxprot &= ~PROT_EXEC;

        /*
         * These checks were added as part of large files.
         *
         * Return ENXIO if the initial position is negative; return EOVERFLOW
         * if (offset + len) would overflow the maximum allowed offset for the
         * type of file descriptor being used.
         */
        if (vp->v_type == VREG) {
                if (pos < 0)
                        return (ENXIO);
                if ((offset_t)len > (OFFSET_MAX(fp) - pos))
                        return (EOVERFLOW);
        }

        if (type == MAP_SHARED && (fp->f_flag & FWRITE) == 0) {
                /* no write access allowed */
                maxprot &= ~PROT_WRITE;
        }

        /*
         * XXX - Do we also adjust maxprot based on protections
         * of the vnode?  E.g. if no execute permission is given
         * on the vnode for the current user, maxprot probably
         * should disallow PROT_EXEC also?  This is different
         * from the write access as this would be a per vnode
         * test as opposed to a per fd test for writability.
         */

        /*
         * Verify that the specified protections are not greater than
         * the maximum allowable protections.  Also test to make sure
         * that the file descriptor does allows for read access since
         * "write only" mappings are hard to do since normally we do
         * the read from the file before the page can be written.
         */
        if (((maxprot & uprot) != uprot) || (fp->f_flag & FREAD) == 0)
                return (EACCES);

        /*
         * If the user specified an address, do some simple checks here
         */
        if ((flags & MAP_FIXED) != 0) {
                caddr_t userlimit;

                /*
                 * Use the user address.  First verify that
                 * the address to be used is page aligned.
                 * Then make some simple bounds checks.
                 */
                if (((uintptr_t)*addrp & PAGEOFFSET) != 0)
                        return (EINVAL);

                userlimit = flags & _MAP_LOW32 ?
                    (caddr_t)USERLIMIT32 : as->a_userlimit;
                switch (valid_usr_range(*addrp, len, uprot, as, userlimit)) {
                case RANGE_OKAY:
                        break;
                case RANGE_BADPROT:
                        return (ENOTSUP);
                case RANGE_BADADDR:
                default:
                        return (ENOMEM);
                }
        }

        if ((prot & (PROT_READ | PROT_WRITE | PROT_EXEC)) &&
            nbl_need_check(vp)) {
                int svmand;
                nbl_op_t nop;

                nbl_start_crit(vp, RW_READER);
                in_crit = 1;
                error = nbl_svmand(vp, fp->f_cred, &svmand);
                if (error != 0)
                        goto done;
                if ((prot & PROT_WRITE) && (type == MAP_SHARED)) {
                        if (prot & (PROT_READ | PROT_EXEC)) {
                                nop = NBL_READWRITE;
                        } else {
                                nop = NBL_WRITE;
                        }
                } else {
                        nop = NBL_READ;
                }
                if (nbl_conflict(vp, nop, 0, LONG_MAX, svmand, NULL)) {
                        error = EACCES;
                        goto done;
                }
        }

        /* discard lwpchan mappings, like munmap() */
        if ((flags & MAP_FIXED) && curproc->p_lcp != NULL)
                lwpchan_delete_mapping(curproc, *addrp, *addrp + len);

        /*
         * Ok, now let the vnode map routine do its thing to set things up.
         */
        error = VOP_MAP(vp, pos, as,
            addrp, len, uprot, maxprot, flags, fp->f_cred, NULL);

        if (error == 0) {
                /*
                 * Tell machine specific code that lwp has mapped shared memory
                 */
                if (flags & MAP_SHARED) {
                        /* EMPTY */
                        LWP_MMODEL_SHARED_AS(*addrp, len);
                }
                if (vp->v_type == VREG &&
                    (flags & (MAP_TEXT | MAP_INITDATA)) != 0) {
                        /*
                         * Mark this as an executable vnode
                         */
                        mutex_enter(&vp->v_lock);
                        vp->v_flag |= VVMEXEC;
```

```
                        mutex_exit(&vp->v_lock);
                }
        }

done:
        if (in_crit)
                nbl_end_crit(vp);
        return (error);
}

#ifdef _LP64
/*
 * LP64 mmap(2) system call: 64-bit offset, 64-bit address.
 *
 * The "large file" mmap routine mmap64(2) is also mapped to this routine
 * by the 64-bit version of libc.
 *
 * Eventually, this should be the only version, and have smmap_common()
 * folded back into it again.  Some day.
 */
caddr_t
smmap64(caddr_t addr, size_t len, int prot, int flags, int fd, off_t pos)
{
        struct file *fp;
        int error;

        if (fd == -1 && (flags & MAP_ANON) != 0)
                error = smmap_common(&addr, len, prot, flags,
                    NULL, (offset_t)pos);
        else if ((fp = getf(fd)) != NULL) {
                error = smmap_common(&addr, len, prot, flags,
                    fp, (offset_t)pos);
                releasef(fd);
        } else
                error = EBADF;

        return (error ? (caddr_t)(uintptr_t)set_errno(error) : addr);
}
#endif  /* _LP64 */

#if defined(_SYSCALL32_IMPL) || defined(_ILP32)

/*
 * ILP32 mmap(2) system call: 32-bit offset, 32-bit address.
 */
caddr_t
smmap32(caddr32_t addr, size32_t len, int prot, int flags, int fd, off32_t pos)
{
        struct file *fp;
        int error;
        caddr_t a = (caddr_t)(uintptr_t)addr;

        if (flags & _MAP_LOW32)
                error = EINVAL;
        else if (fd == -1 && (flags & MAP_ANON) != 0)
                error = smmap_common(&a, (size_t)len, prot,
                    flags | _MAP_LOW32, NULL, (offset_t)pos);
        else if ((fp = getf(fd)) != NULL) {
                error = smmap_common(&a, (size_t)len, prot,
                    flags | _MAP_LOW32, fp, (offset_t)pos);
                releasef(fd);
        } else
                error = EBADF;

        ASSERT(error != 0 || (uintptr_t)(a + len) < (uintptr_t)UINT32_MAX);
```

```c
        return (error ? (caddr_t)(uintptr_t)set_errno(error) : a);
}

/*
 * ILP32 mmap64(2) system call: 64-bit offset, 32-bit address.
 *
 * Now things really get ugly because we can’t use the C-style
 * calling convention for more than 6 args, and 64-bit parameter
 * passing on 32-bit systems is less than clean.
 */

struct mmaplf32a {
        caddr_t addr;
        size_t len;
#ifdef _LP64
        /*
         * 32-bit contents, 64-bit cells
         */
        uint64_t prot;
        uint64_t flags;
        uint64_t fd;
        uint64_t offhi;
        uint64_t offlo;
#else
        /*
         * 32-bit contents, 32-bit cells
         */
        uint32_t prot;
        uint32_t flags;
        uint32_t fd;
        uint32_t offhi;
        uint32_t offlo;
#endif
};

int
smmaplf32(struct mmaplf32a *uap, rval_t *rvp)
{
        struct file *fp;
        int error;
        caddr_t a = uap->addr;
        int flags = (int)uap->flags;
        int fd = (int)uap->fd;
#ifdef _BIG_ENDIAN
        offset_t off = ((u_offset_t)uap->offhi << 32) | (u_offset_t)uap->offlo;
#else
        offset_t off = ((u_offset_t)uap->offlo << 32) | (u_offset_t)uap->offhi;
#endif

        if (flags & _MAP_LOW32)
                error = EINVAL;
        else if (fd == -1 && (flags & MAP_ANON) != 0)
                error = smmap_common(&a, uap->len, (int)uap->prot,
                    flags | _MAP_LOW32, NULL, off);
        else if ((fp = getf(fd)) != NULL) {
                error = smmap_common(&a, uap->len, (int)uap->prot,
                    flags | _MAP_LOW32, fp, off);
                releasef(fd);
        } else
                error = EBADF;

        if (error == 0)
                rvp->r_val1 = (uintptr_t)a;
        return (error);
}
```

```c
#endif  /* _SYSCALL32_IMPL || _ILP32 */

int
munmap(caddr_t addr, size_t len)
{
        struct proc *p = curproc;
        struct as *as = p->p_as;

        if (((uintptr_t)addr & PAGEOFFSET) != 0 || len == 0)
                return (set_errno(EINVAL));

        if (valid_usr_range(addr, len, 0, as, as->a_userlimit) != RANGE_OKAY)
                return (set_errno(EINVAL));

        /*
         * Discard lwpchan mappings.
         */
        if (p->p_lcp != NULL)
                lwpchan_delete_mapping(p, addr, addr + len);
        if (as_unmap(as, addr, len) != 0)
                return (set_errno(EINVAL));

        return (0);
}

int
mprotect(caddr_t addr, size_t len, int prot)
{
        struct as *as = curproc->p_as;
        uint_t uprot = prot | PROT_USER;
        int error;

        if (((uintptr_t)addr & PAGEOFFSET) != 0 || len == 0)
                return (set_errno(EINVAL));

        switch (valid_usr_range(addr, len, prot, as, as->a_userlimit)) {
        case RANGE_OKAY:
                break;
        case RANGE_BADPROT:
                return (set_errno(ENOTSUP));
        case RANGE_BADADDR:
        default:
                return (set_errno(ENOMEM));
        }

        error = as_setprot(as, addr, len, uprot);
        if (error)
                return (set_errno(error));
        return (0);
}

#define MC_CACHE        128                     /* internal result buffer */
#define MC_QUANTUM      (MC_CACHE * PAGESIZE)   /* addresses covered in loop */

int
mincore(caddr_t addr, size_t len, char *vecp)
{
        struct as *as = curproc->p_as;
        caddr_t ea;                     /* end address of loop */
        size_t rl;                      /* inner result length */
        char vec[MC_CACHE];             /* local vector cache */
        int error;
        model_t model;
        long    llen;

        model = get_udatamodel();
        /*
         * Validate form of address parameters.
         */
        if (model == DATAMODEL_NATIVE) {
                llen = (long)len;
        } else {
                llen = (int32_t)(size32_t)len;
        }
        if (((uintptr_t)addr & PAGEOFFSET) != 0 || llen <= 0)
                return (set_errno(EINVAL));

        if (valid_usr_range(addr, len, 0, as, as->a_userlimit) != RANGE_OKAY)
                return (set_errno(ENOMEM));

        /*
         * Loop over subranges of interval [addr : addr + len), recovering
         * results internally and then copying them out to caller.  Subrange
         * is based on the size of MC_CACHE, defined above.
         */
        for (ea = addr + len; addr < ea; addr += MC_QUANTUM) {
                error = as_incore(as, addr,
                    (size_t)MIN(MC_QUANTUM, ea - addr), vec, &rl);
                if (rl != 0) {
                        rl = (rl + PAGESIZE - 1) / PAGESIZE;
                        if (copyout(vec, vecp, rl) != 0)
                                return (set_errno(EFAULT));
                        vecp += rl;
                }
                if (error != 0)
                        return (set_errno(ENOMEM));
        }
        return (0);
}
```