```
**********************************************************
   23757 Sun Jul 14 17:37:58 2013
new/usr/src/cmd/cmd-crypto/digest/digest.c
XXX Enlarge data buffer in digest/mac to boost performance
**********************************************************
```

```
/*
 * 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 2010 Sun Microsystems, Inc.  All rights reserved.
 * Use is subject to license terms.
 */

/*
 * digest.c
 *
 * Implements digest(1) and mac(1) commands
 * If command name is mac, performs mac operation
 * else perform digest operation
 *
 * See the man pages for digest and mac for details on
 * how these commands work.
 */

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <ctype.h>
#include <strings.h>
#include <libintl.h>
#include <libgen.h>
#include <locale.h>
#include <errno.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <security/cryptoki.h>
#include <limits.h>
#include <cryptoutil.h>
#include <kmfapi.h>

/*
 * Buffer size for reading file. This is given a rather high value
 * to get better performance when a hardware provider is present.
 */
#define BUFFERSIZE      (1024 * 64)
#define BUFFERSIZE      (4096)          /* Buffer size for reading file */

/*
```



```
 * RESULTLEN - large enough size in bytes to hold result for
 * digest and mac results for all mechanisms
 */
#define RESULTLEN       (512)

/*
 * Exit Status codes
 */
#ifndef EXIT_SUCCESS
#define EXIT_SUCCESS    0       /* No errors */
#define EXIT_FAILURE    1       /* All errors except usage */
#endif /* EXIT_SUCCESS */

#define EXIT_USAGE      2       /* usage/syntax error */

#define MAC_NAME        "mac"           /* name of mac command */
#define MAC_OPTIONS     "lva:k:T:K:"    /* for getopt */
#define DIGEST_NAME     "digest"        /* name of digest command */
#define DIGEST_OPTIONS  "lva:"          /* for getopt */

/* Saved command line options */
static boolean_t vflag = B_FALSE;       /* -v (verbose) flag, optional */
static boolean_t aflag = B_FALSE;       /* -a <algorithm> flag, required */
static boolean_t lflag = B_FALSE;       /* -l flag, for mac and digest */
static boolean_t kflag = B_FALSE;       /* -k keyfile */
static boolean_t Tflag = B_FALSE;       /* -T token_spec */
static boolean_t Kflag = B_FALSE;       /* -K key_label */

static char *keyfile = NULL;     /* name of file containing key value */
static char *token_label = NULL; /* tokensSpec: tokenName[:manufId[:serial]] */
static char *key_label = NULL;   /* PKCS#11 symmetric token key label */

static CK_BYTE buf[BUFFERSIZE];

struct mech_alias {
        CK_MECHANISM_TYPE type;
        char *alias;
        CK_ULONG keysize_min;
        CK_ULONG keysize_max;
        int keysize_unit;
        boolean_t available;
};
```

*unchanged_portion_omitted_*