FreeBSD kernel opencrypto code Reference Manual

Generated by Doxygen 1.4.7

Sat Feb 24 20:08:24 2007

Contents

1 FreeBSD kernel opencrypto code Main Page 1

2 FreeBSD kernel opencrypto code Directory Hierarchy 3

2.1 FreeBSD kernel opencrypto code Directories 3

3 FreeBSD kernel opencrypto code Data Structure Index 5

3.1 FreeBSD kernel opencrypto code Data Structures 5

4 FreeBSD kernel opencrypto code File Index 7

4.1 FreeBSD kernel opencrypto code File List 7

5 FreeBSD kernel opencrypto code Directory Documentation 9

5.1 /usr/src/sys/opencrypto/ Directory Reference 9

5.2 /usr/src/ Directory Reference 10

5.3 /usr/src/sys/ Directory Reference 11

5.4 /usr/ Directory Reference 12

6 FreeBSD kernel opencrypto code Data Structure Documentation 13

6.1 auth_hash Struct Reference 13

6.2 authctx Union Reference 15

6.3 cast_key Struct Reference 17

6.4 comp_algo Struct Reference 18

6.5 crparam Struct Reference 19

6.6 crypt_kop Struct Reference 20

6.7 crypt_op Struct Reference 22

6.8 cryptkop Struct Reference 24

6.9 cryptocap Struct Reference 26

6.10 cryptodesc Struct Reference 29

6.11 cryptoini Struct Reference 31

6.12 cryptop Struct Reference 33

6.13 cryptostats Struct Reference 36
6.14 cryptotstat Struct Reference 39
6.15 csession Struct Reference 40
6.16 deflate_buf Struct Reference 41
6.17 enc_xform Struct Reference 42
6.18 fcrypt Struct Reference 44
6.19 RMD160Context Struct Reference 45
6.20 session_op Struct Reference 46
6.21 swcr_data Struct Reference 48

7 FreeBSD kernel opencrypto code File Documentation 51

7.1 notreviewed.dox File Reference 51
7.2 /usr/src/sys/opencrypto/cast.c File Reference 52
7.3 /usr/src/sys/opencrypto/cast.h File Reference 55
7.4 /usr/src/sys/opencrypto/castsb.h File Reference 56
7.5 /usr/src/sys/opencrypto/criov.c File Reference 58
7.6 /usr/src/sys/opencrypto/crypto.c File Reference 61
7.7 /usr/src/sys/opencrypto/crypto_if.m File Reference 73
7.8 /usr/src/sys/opencrypto/cryptodev.c File Reference 74
7.9 /usr/src/sys/opencrypto/cryptodev.h File Reference 82
7.10 /usr/src/sys/opencrypto/cryptosoft.c File Reference 103
7.11 /usr/src/sys/opencrypto/cryptosoft.h File Reference 110
7.12 /usr/src/sys/opencrypto/deflate.c File Reference 112
7.13 /usr/src/sys/opencrypto/deflate.h File Reference 114
7.14 /usr/src/sys/opencrypto/rmd160.c File Reference 116
7.15 /usr/src/sys/opencrypto/rmd160.h File Reference 122
7.16 /usr/src/sys/opencrypto/skipjack.c File Reference 124
7.17 /usr/src/sys/opencrypto/skipjack.h File Reference 128
7.18 /usr/src/sys/opencrypto/xform.c File Reference 129
7.19 /usr/src/sys/opencrypto/xform.h File Reference 142

Chapter 1

FreeBSD kernel opencrypto code Main Page

IMPORTANT: This API documentation may contain both functions which are public and functions that are for internal use only. Since we have not reviewed every part of the documentation yet, *some internal functions are not marked as such*. Until we finish reviewing the API documentation and add appropriate comments to functions which are only for internal use, you should take this into account. In case you want to use a function of this kernel subsystem in another kernel subsystem you should search for precedence of use outside this subsystem. If the function is not used outside this subsystem you should ask on the mailinglists about it, else you risk breaking something.

Chapter 2

FreeBSD kernel opencrypto code Directory Hierarchy

2.1 FreeBSD kernel opencrypto code Directories

This directory hierarchy is sorted roughly, but not completely, alphabetically:

- usr 12
  - src 10
    - sys 11
      - opencrypto 9

Chapter 3

FreeBSD kernel opencrypto code Data Structure Index

3.1 FreeBSD kernel opencrypto code Data Structures

Here are the data structures with brief descriptions:

auth_hash 13
authctx 15
cast_key 17
comp_algo 18
crparam 19
crypt_kop 20
crypt_op 22
cryptkop 24
cryptocap 26
cryptodesc 29
cryptoini 31
cryptop 33
cryptostats 36
cryptotstat 39
csession 40
deflate_buf 41
enc_xform 42
fcrypt 44
RMD160Context 45
session_op 46
swcr_data 48

Chapter 4

FreeBSD kernel opencrypto code File Index

4.1 FreeBSD kernel opencrypto code File List

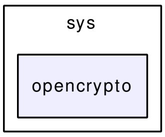
Here is a list of all files with brief descriptions:

/usr/src/sys/opencrypto/cast.c 52
/usr/src/sys/opencrypto/cast.h 55
/usr/src/sys/opencrypto/castsb.h 56
/usr/src/sys/opencrypto/criov.c 58
/usr/src/sys/opencrypto/crypto.c 61
/usr/src/sys/opencrypto/crypto_if.m 73
/usr/src/sys/opencrypto/cryptodev.c 74
/usr/src/sys/opencrypto/cryptodev.h 82
/usr/src/sys/opencrypto/cryptosoft.c 103
/usr/src/sys/opencrypto/cryptosoft.h 110
/usr/src/sys/opencrypto/deflate.c 112
/usr/src/sys/opencrypto/deflate.h 114
/usr/src/sys/opencrypto/rmd160.c 116
/usr/src/sys/opencrypto/rmd160.h 122
/usr/src/sys/opencrypto/skipjack.c 124
/usr/src/sys/opencrypto/skipjack.h 128
/usr/src/sys/opencrypto/xform.c 129
/usr/src/sys/opencrypto/xform.h 142

Chapter 5

FreeBSD kernel opencrypto code Directory Documentation

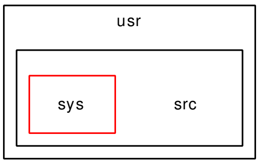
5.1 /usr/src/sys/opencrypto/ Directory Reference



Files

- file cast.c
- file cast.h
- file castsb.h
- file criov.c
- file crypto.c
- file crypto_if.m
- file cryptodev.c
- file cryptodev.h
- file cryptosoft.c
- file cryptosoft.h
- file deflate.c
- file deflate.h
- file rmd160.c
- file rmd160.h
- file skipjack.c
- file skipjack.h
- file xform.c
- file xform.h

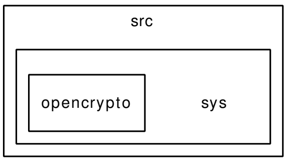
5.2 /usr/src/ Directory Reference



Directories

- directory sys

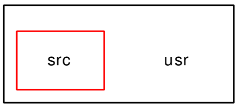
5.3 /usr/src/sys/ Directory Reference



Directories

- directory opencrypto

5.4 /usr/ Directory Reference



Directories

- directory src

Chapter 6

FreeBSD kernel opencrypto code Data Structure Documentation

6.1 auth_hash Struct Reference

```
#include <xform.h>
```

Data Fields

- int type
- char ∗ name
- u_int16_t keysize
- u_int16_t hashsize
- u_int16_t blocksize
- u_int16_t ctxsize
- void(∗ Init)(void ∗)
- int(∗ Update)(void ∗, u_int8_t ∗, u_int16_t)
- void(∗ Final)(u_int8_t ∗, void ∗)

6.1.1 Detailed Description

Definition at line 34 of file xform.h.

6.1.2 Field Documentation

6.1.2.1 u_int16_t auth_hash::blocksize

Definition at line 39 of file xform.h.

Referenced by swcr_authprepare().

6.1.2.2 u_int16_t auth_hash::ctxsize

Definition at line 40 of file xform.h.

Referenced by swcr_freesession(), and swcr_newsession().

6.1.2.3 void(∗ auth_hash::Final)(u_int8_t ∗, void ∗)

Referenced by swcr_authprepare().

6.1.2.4 u_int16_t auth_hash::hashsize

Definition at line 38 of file xform.h.

6.1.2.5 void(∗ auth_hash::Init)(void ∗)

Referenced by swcr_authprepare(), and swcr_newsession().

6.1.2.6 u_int16_t auth_hash::keysize

Definition at line 37 of file xform.h.

Referenced by cryptof_ioctl().

6.1.2.7 char∗ auth_hash::name

Definition at line 36 of file xform.h.

6.1.2.8 int auth_hash::type

Definition at line 35 of file xform.h.

Referenced by cryptof_ioctl(), and swcr_authprepare().

6.1.2.9 int(∗ auth_hash::Update)(void ∗, u_int8_t ∗, u_int16_t)

Referenced by swcr_authprepare().

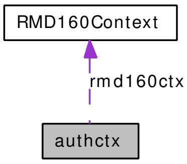
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/xform.h

6.2 authctx Union Reference

```
#include <xform.h>
```

Collaboration diagram for authctx:



Data Fields

- MD5_CTX md5ctx
- SHA1_CTX sha1ctx
- RMD160_CTX rmd160ctx
- SHA256_CTX sha256ctx
- SHA384_CTX sha384ctx
- SHA512_CTX sha512ctx

6.2.1 Detailed Description

Definition at line 67 of file xform.h.

6.2.2 Field Documentation

6.2.2.1 MD5_CTX authctx::md5ctx

Definition at line 68 of file xform.h.

6.2.2.2 RMD160_CTX authctx::rmd160ctx

Definition at line 70 of file xform.h.

6.2.2.3 SHA1_CTX authctx::sha1ctx

Definition at line 69 of file xform.h.

6.2.2.4 SHA256_CTX authctx::sha256ctx

Definition at line 71 of file xform.h.

6.2.2.5 SHA384_CTX authctx::sha384ctx

Definition at line 72 of file xform.h.

6.2.2.6 SHA512_CTX authctx::sha512ctx

Definition at line 73 of file xform.h.

The documentation for this union was generated from the following file:

- /usr/src/sys/opencrypto/xform.h

6.3 cast_key Struct Reference

```
#include <cast.h>
```

Data Fields

- u_int32_t xkey [32]
- int rounds

6.3.1 Detailed Description

Definition at line 14 of file cast.h.

6.3.2 Field Documentation

6.3.2.1 int cast_key::rounds

Definition at line 16 of file cast.h.

Referenced by cast_decrypt(), cast_encrypt(), and cast_setkey().

6.3.2.2 u_int32_t cast_key::xkey[32]

Definition at line 15 of file cast.h.

Referenced by cast_setkey().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cast.h

6.4 comp_algo Struct Reference

```
#include <xform.h>
```

Data Fields

- int type
- char ∗ name
- size_t minlen
- u_int32_t(∗ compress)(u_int8_t ∗, u_int32_t, u_int8_t ∗∗)
- u_int32_t(∗ decompress)(u_int8_t ∗, u_int32_t, u_int8_t ∗∗)

6.4.1 Detailed Description

Definition at line 59 of file xform.h.

6.4.2 Field Documentation

6.4.2.1 u_int32_t(∗ comp_algo::compress)(u_int8_t ∗, u_int32_t, u_int8_t ∗∗)

Referenced by swcr_compdec().

6.4.2.2 u_int32_t(∗ comp_algo::decompress)(u_int8_t ∗, u_int32_t, u_int8_t ∗∗)

Referenced by swcr_compdec().

6.4.2.3 size_t comp_algo::minlen

Definition at line 62 of file xform.h.

6.4.2.4 char∗ comp_algo::name

Definition at line 61 of file xform.h.

6.4.2.5 int comp_algo::type

Definition at line 60 of file xform.h.

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/xform.h

6.5 crparam Struct Reference

```
#include <cryptodev.h>
```

Data Fields

- caddr_t crp_p
- u_int crp_nbits

6.5.1 Detailed Description

Definition at line 156 of file cryptodev.h.

6.5.2 Field Documentation

6.5.2.1 u_int crparam::crp_nbits

Definition at line 158 of file cryptodev.h.

Referenced by cryptodev_key().

6.5.2.2 caddr_t crparam::crp_p

Definition at line 157 of file cryptodev.h.

Referenced by cryptodev_key().

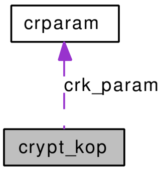
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.6 crypt_kop Struct Reference

```
#include <cryptodev.h>
```

Collaboration diagram for crypt_kop:



Data Fields

- u_int crk_op
- u_int crk_status
- u_short crk_iparams
- u_short crk_oparams
- u_int crk_pad1
- crparam crk_param [CRK_MAXPARAM]

6.6.1 Detailed Description

Definition at line 163 of file cryptodev.h.

6.6.2 Field Documentation

6.6.2.1 u_short crypt_kop::crk_iparams

Definition at line 166 of file cryptodev.h.

Referenced by cryptodev_key().

6.6.2.2 u_int crypt_kop::crk_op

Definition at line 164 of file cryptodev.h.

Referenced by cryptodev_key().

6.6.2.3 u_short crypt_kop::crk_oparams

Definition at line 167 of file cryptodev.h.

Referenced by cryptodev_key().

6.6.2.4 u_int crypt_kop::crk_pad1

Definition at line 168 of file cryptodev.h.

6.6.2.5 struct crparam crypt_kop::crk_param[CRK_MAXPARAM]

Definition at line 169 of file cryptodev.h.

Referenced by cryptodev_key().

6.6.2.6 u_int crypt_kop::crk_status

Definition at line 165 of file cryptodev.h.

Referenced by cryptodev_key().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.7 crypt_op Struct Reference

```
#include <cryptodev.h>
```

Data Fields

- u_int32_t ses
- u_int16_t op
- u_int16_t flags
- u_int len
- caddr_t src
- caddr_t dst
- caddr_t mac
- caddr_t iv

6.7.1 Detailed Description

Definition at line 142 of file cryptodev.h.

6.7.2 Field Documentation

6.7.2.1 caddr_t crypt_op::dst

Definition at line 150 of file cryptodev.h.

Referenced by cryptodev_op().

6.7.2.2 u_int16_t crypt_op::flags

Definition at line 147 of file cryptodev.h.

Referenced by cryptodev_op().

6.7.2.3 caddr_t crypt_op::iv

Definition at line 152 of file cryptodev.h.

Referenced by cryptodev_op().

6.7.2.4 u_int crypt_op::len

Definition at line 149 of file cryptodev.h.

Referenced by cryptodev_op().

6.7.2.5 caddr_t crypt_op::mac

Definition at line 151 of file cryptodev.h.

Referenced by cryptodev_op().

6.7.2.6 u_int16_t crypt_op::op

Definition at line 144 of file cryptodev.h.

Referenced by cryptodev_op().

6.7.2.7 u_int32_t crypt_op::ses

Definition at line 143 of file cryptodev.h.

Referenced by cryptof_ioctl().

6.7.2.8 caddr_t crypt_op::src

Definition at line 150 of file cryptodev.h.

Referenced by cryptodev_op().

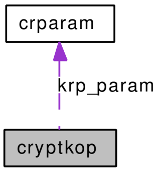
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.8 cryptkop Struct Reference

```
#include <cryptodev.h>
```

Collaboration diagram for cryptkop:



Public Member Functions

- TAILQ_ENTRY (cryptkop) krp_next

Data Fields

- u_int krp_op
- u_int krp_status
- u_short krp_iparams
- u_short krp_oparams
- u_int32_t krp_hid
- crparam krp_param [CRK_MAXPARAM]
- int(∗ krp_callback)(struct cryptkop ∗)

6.8.1 Detailed Description

Definition at line 312 of file cryptodev.h.

6.8.2 Member Function Documentation

6.8.2.1 cryptkop::TAILQ_ENTRY (cryptkop)

6.8.3 Field Documentation

6.8.3.1 int(∗ cryptkop::krp_callback)(struct cryptkop ∗)

Referenced by crypto_kinvoke(), and crypto_ret_proc().

6.8.3.2 u_int32_t cryptkop::krp_hid

Definition at line 319 of file cryptodev.h.

Referenced by crypto_kdone(), and crypto_proc().

6.8.3.3 u_short cryptkop::krp_iparams

Definition at line 317 of file cryptodev.h.

Referenced by cryptodev_key().

6.8.3.4 u_int cryptkop::krp_op

Definition at line 315 of file cryptodev.h.

Referenced by crypto_kinvoke().

6.8.3.5 u_short cryptkop::krp_oparams

Definition at line 318 of file cryptodev.h.

6.8.3.6 struct crparam cryptkop::krp_param[CRK_MAXPARAM]

Definition at line 320 of file cryptodev.h.

6.8.3.7 u_int cryptkop::krp_status

Definition at line 316 of file cryptodev.h.

Referenced by crypto_kdone().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.9 cryptocap Struct Reference

```
#include <cryptodev.h>
```

Data Fields

- u_int32_t cc_sessions
- u_int32_t cc_koperations
- u_int16_t cc_max_op_len [CRYPTO_ALGORITHM_MAX+1]
- u_int8_t cc_alg [CRYPTO_ALGORITHM_MAX+1]
- u_int8_t cc_kalg [CRK_ALGORITHM_MAX+1]
- u_int8_t cc_flags
- u_int8_t cc_qblocked
- u_int8_t cc_kqblocked
- void ∗ cc_arg
- int(∗ cc_newsession)(void ∗, u_int32_t ∗, struct cryptoini ∗)
- int(∗ cc_process)(void ∗, struct cryptop ∗, int)
- int(∗ cc_freesession)(void ∗, u_int64_t)
- void ∗ cc_karg
- int(∗ cc_kprocess)(void ∗, struct cryptkop ∗, int)

6.9.1 Detailed Description

Definition at line 332 of file cryptodev.h.

6.9.2 Field Documentation

6.9.2.1 u_int8_t cryptocap::cc_alg[CRYPTO_ALGORITHM_MAX+1]

Definition at line 342 of file cryptodev.h.

Referenced by crypto_newsession(), crypto_register(), crypto_unregister(), and crypto_unregister_all().

6.9.2.2 void∗ cryptocap::cc_arg

Definition at line 353 of file cryptodev.h.

Referenced by crypto_freesession(), crypto_invoke(), crypto_newsession(), and crypto_register().

6.9.2.3 u_int8_t cryptocap::cc_flags

Definition at line 346 of file cryptodev.h.

Referenced by crypto_freesession(), crypto_get_driverid(), crypto_getfeat(), crypto_invoke(), crypto_kdone(), crypto_kinvoke(), and crypto_newsession().

6.9.2.4 int(∗ cryptocap::cc_freesession)(void ∗, u_int64_t)

Referenced by crypto_freesession(), and crypto_register().

6.9.2.5 u_int8_t cryptocap::cc_kalg[CRK_ALGORITHM_MAX+1]

Definition at line 344 of file cryptodev.h.

Referenced by crypto_getfeat(), crypto_kinvoke(), and crypto_kregister().

6.9.2.6 void∗ cryptocap::cc_karg

Definition at line 357 of file cryptodev.h.

Referenced by crypto_kregister().

6.9.2.7 u_int32_t cryptocap::cc_koperations

Definition at line 334 of file cryptodev.h.

Referenced by crypto_kdone(), crypto_remove(), crypto_unregister(), and crypto_unregister_all().

6.9.2.8 int(∗ cryptocap::cc_kprocess)(void ∗, struct cryptkop ∗, int)

Referenced by crypto_getfeat(), crypto_kinvoke(), crypto_kregister(), and crypto_proc().

6.9.2.9 u_int8_t cryptocap::cc_kqblocked

Definition at line 351 of file cryptodev.h.

Referenced by crypto_kinvoke(), crypto_proc(), and crypto_unblock().

6.9.2.10 u_int16_t cryptocap::cc_max_op_len[CRYPTO_ALGORITHM_MAX+1]

Definition at line 340 of file cryptodev.h.

Referenced by crypto_register(), crypto_unregister(), and crypto_unregister_all().

6.9.2.11 int(∗ cryptocap::cc_newsession)(void ∗, u_int32_t ∗, struct cryptoini ∗)

Referenced by crypto_newsession(), and crypto_register().

6.9.2.12 int(∗ cryptocap::cc_process)(void ∗, struct cryptop ∗, int)

Referenced by crypto_get_driverid(), crypto_invoke(), crypto_proc(), and crypto_register().

6.9.2.13 u_int8_t cryptocap::cc_qblocked

Definition at line 350 of file cryptodev.h.

Referenced by crypto_dispatch(), crypto_proc(), and crypto_unblock().

6.9.2.14 u_int32_t cryptocap::cc_sessions

Definition at line 333 of file cryptodev.h.

Referenced by crypto_freesession(), crypto_get_driverid(), crypto_newsession(), crypto_register(), crypto_remove(), crypto_unregister(), and crypto_unregister_all().

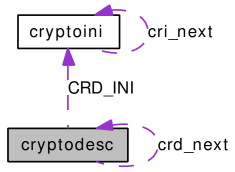
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.10 cryptodesc Struct Reference

```
#include <cryptodev.h>
```

Collaboration diagram for cryptodesc:



Data Fields

- int crd_skip
- int crd_len
- int crd_inject
- int crd_flags
- cryptoini CRD_INI
- cryptodesc ∗ crd_next

6.10.1 Detailed Description

Definition at line 240 of file cryptodev.h.

6.10.2 Field Documentation

6.10.2.1 int cryptodesc::crd_flags

Definition at line 244 of file cryptodev.h.

Referenced by cryptodev_op(), swcr_authcompute(), swcr_compdec(), and swcr_encdec().

6.10.2.2 struct cryptoini cryptodesc::CRD_INI

Definition at line 254 of file cryptodev.h.

Referenced by crypto_invoke().

6.10.2.3 int cryptodesc::crd_inject

Definition at line 243 of file cryptodev.h.

Referenced by cryptodev_op(), swcr_authcompute(), and swcr_encdec().

6.10.2.4 int cryptodesc::crd_len

Definition at line 242 of file cryptodev.h.

Referenced by cryptodev_op(), swcr_authcompute(), swcr_compdec(), and swcr_encdec().

6.10.2.5 struct cryptodesc∗ cryptodesc::crd_next

Definition at line 260 of file cryptodev.h.

Referenced by crypto_freereq(), crypto_getreq(), crypto_invoke(), cryptodev_op(), and swcr_process().

6.10.2.6 int cryptodesc::crd_skip

Definition at line 241 of file cryptodev.h.

Referenced by cryptodev_op(), swcr_authcompute(), swcr_compdec(), and swcr_encdec().

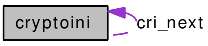
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.11 cryptoini Struct Reference

`#include <cryptodev.h>`

Collaboration diagram for cryptoini:



Data Fields

- int cri_alg
- int cri_klen
- int cri_mlen
- caddr_t cri_key
- u_int8_t cri_iv [EALG_MAX_BLOCK_LEN]
- cryptoini ∗ cri_next

6.11.1 Detailed Description

Definition at line 229 of file cryptodev.h.

6.11.2 Field Documentation

6.11.2.1 int cryptoini::cri_alg

Definition at line 230 of file cryptodev.h.

Referenced by crypto_newsession(), and swcr_newsession().

6.11.2.2 u_int8_t cryptoini::cri_iv[EALG_MAX_BLOCK_LEN]

Definition at line 235 of file cryptodev.h.

6.11.2.3 caddr_t cryptoini::cri_key

Definition at line 234 of file cryptodev.h.

Referenced by swcr_newsession().

6.11.2.4 int cryptoini::cri_klen

Definition at line 231 of file cryptodev.h.

Referenced by swcr_newsession().

6.11.2.5 int cryptoini::cri_mlen

Definition at line 232 of file cryptodev.h.

Referenced by swcr_newsession().

6.11.2.6 struct cryptoini∗ cryptoini::cri_next

Definition at line 236 of file cryptodev.h.

Referenced by crypto_invoke(), crypto_newsession(), and swcr_newsession().

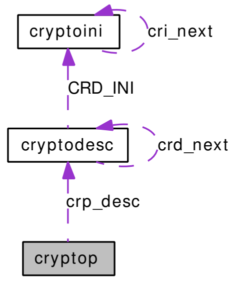
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.12 cryptop Struct Reference

```
#include <cryptodev.h>
```

Collaboration diagram for cryptop:



Public Member Functions

- TAILQ_ENTRY (cryptop) crp_next

Data Fields

- u_int64_t crp_sid
- int crp_ilen
- int crp_olen
- int crp_etype
- int crp_flags
- caddr_t crp_buf
- caddr_t crp_opaque
- cryptodesc ∗ crp_desc
- int(∗ crp_callback)(struct cryptop ∗)
- bintime crp_tstamp

6.12.1 Detailed Description

Definition at line 264 of file cryptodev.h.

6.12.2 Member Function Documentation

6.12.2.1 cryptop::TAILQ_ENTRY (cryptop)

6.12.3 Field Documentation

6.12.3.1 caddr_t cryptop::crp_buf

Definition at line 291 of file cryptodev.h.

Referenced by cryptodev_op(), and swcr_process().

6.12.3.2 int(∗ cryptop::crp_callback)(struct cryptop ∗)

Referenced by crypto_done(), crypto_invoke(), crypto_ret_proc(), and cryptodev_op().

6.12.3.3 struct cryptodesc∗ cryptop::crp_desc

Definition at line 293 of file cryptodev.h.

Referenced by crypto_freereq(), crypto_getreq(), crypto_invoke(), cryptodev_op(), and swcr_process().

6.12.3.4 int cryptop::crp_etype

Definition at line 271 of file cryptodev.h.

Referenced by crypto_done(), crypto_invoke(), cryptodev_cb(), cryptodev_op(), and swcr_process().

6.12.3.5 int cryptop::crp_flags

Definition at line 281 of file cryptodev.h.

Referenced by crypto_dispatch(), crypto_done(), crypto_proc(), cryptodev_op(), and swcr_process().

6.12.3.6 int cryptop::crp_ilen

Definition at line 268 of file cryptodev.h.

Referenced by cryptodev_op().

6.12.3.7 int cryptop::crp_olen

Definition at line 269 of file cryptodev.h.

Referenced by swcr_process().

6.12.3.8 caddr_t cryptop::crp_opaque

Definition at line 292 of file cryptodev.h.

Referenced by cryptodev_cb(), and cryptodev_op().

6.12.3.9 u_int64_t cryptop::crp_sid

Definition at line 267 of file cryptodev.h.

Referenced by crypto_dispatch(), crypto_done(), crypto_invoke(), crypto_proc(), cryptodev_op(), and swcr_process().

6.12.3.10 struct bintime cryptop::crp_tstamp

Definition at line 297 of file cryptodev.h.

Referenced by crypto_dispatch(), crypto_done(), crypto_invoke(), and crypto_ret_proc().

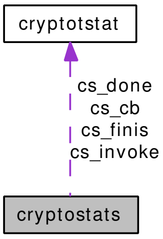
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.13 cryptostats Struct Reference

`#include <cryptodev.h>`

Collaboration diagram for cryptostats:



Data Fields

- u_int32_t cs_ops
- u_int32_t cs_errs
- u_int32_t cs_kops
- u_int32_t cs_kerrs
- u_int32_t cs_intrs
- u_int32_t cs_rets
- u_int32_t cs_blocks
- u_int32_t cs_kblocks
- cryptotstat cs_invoke
- cryptotstat cs_done
- cryptotstat cs_cb
- cryptotstat cs_finis

6.13.1 Detailed Description

Definition at line 206 of file cryptodev.h.

6.13.2 Field Documentation

6.13.2.1 u_int32_t cryptostats::cs_blocks

Definition at line 213 of file cryptodev.h.

Referenced by crypto_proc().

6.13.2.2 struct cryptotstat cryptostats::cs_cb

Definition at line 223 of file cryptodev.h.

Referenced by crypto_done(), and crypto_ret_proc().

6.13.2.3 struct cryptotstat cryptostats::cs_done

Definition at line 222 of file cryptodev.h.

Referenced by crypto_done().

6.13.2.4 u_int32_t cryptostats::cs_errs

Definition at line 208 of file cryptodev.h.

Referenced by crypto_done().

6.13.2.5 struct cryptotstat cryptostats::cs_finis

Definition at line 224 of file cryptodev.h.

Referenced by crypto_done(), and crypto_ret_proc().

6.13.2.6 u_int32_t cryptostats::cs_intrs

Definition at line 211 of file cryptodev.h.

Referenced by crypto_proc().

6.13.2.7 struct cryptotstat cryptostats::cs_invoke

Definition at line 221 of file cryptodev.h.

Referenced by crypto_invoke().

6.13.2.8 u_int32_t cryptostats::cs_kblocks

Definition at line 214 of file cryptodev.h.

Referenced by crypto_proc().

6.13.2.9 u_int32_t cryptostats::cs_kerrs

Definition at line 210 of file cryptodev.h.

Referenced by crypto_kdone().

6.13.2.10 u_int32_t cryptostats::cs_kops

Definition at line 209 of file cryptodev.h.

Referenced by crypto_kdispatch().

6.13.2.11 u_int32_t cryptostats::cs_ops

Definition at line 207 of file cryptodev.h.

Referenced by crypto_dispatch().

6.13.2.12 u_int32_t cryptostats::cs_rets

Definition at line 212 of file cryptodev.h.

Referenced by crypto_ret_proc().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.14 cryptotstat Struct Reference

```
#include <cryptodev.h>
```

Data Fields

- timespec acc
- timespec min
- timespec max
- u_int32_t count

6.14.1 Detailed Description

Definition at line 199 of file cryptodev.h.

6.14.2 Field Documentation

6.14.2.1 struct timespec cryptotstat::acc

Definition at line 200 of file cryptodev.h.

Referenced by crypto_tstat().

6.14.2.2 u_int32_t cryptotstat::count

Definition at line 203 of file cryptodev.h.

Referenced by crypto_tstat().

6.14.2.3 struct timespec cryptotstat::max

Definition at line 202 of file cryptodev.h.

Referenced by crypto_tstat().

6.14.2.4 struct timespec cryptotstat::min

Definition at line 201 of file cryptodev.h.

Referenced by crypto_tstat().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.15 csession Struct Reference

6.15.1 Detailed Description

Definition at line 57 of file cryptodev.c.

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.c

6.16 deflate_buf Struct Reference

```
#include <deflate.h>
```

Data Fields

- u_int8_t ∗ out
- u_int32_t size
- int flag

6.16.1 Detailed Description

Definition at line 50 of file deflate.h.

6.16.2 Field Documentation

6.16.2.1 int deflate_buf::flag

Definition at line 53 of file deflate.h.

Referenced by deflate_global().

6.16.2.2 u_int8_t∗ deflate_buf::out

Definition at line 51 of file deflate.h.

Referenced by deflate_global().

6.16.2.3 u_int32_t deflate_buf::size

Definition at line 52 of file deflate.h.

Referenced by deflate_global().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/deflate.h

6.17 enc_xform Struct Reference

```
#include <xform.h>
```

Data Fields

- int type
- char ∗ name
- u_int16_t blocksize
- u_int16_t minkey
- u_int16_t maxkey
- void(∗ encrypt)(caddr_t, u_int8_t ∗)
- void(∗ decrypt)(caddr_t, u_int8_t ∗)
- int(∗ setkey)(u_int8_t ∗∗, u_int8_t ∗, int len)
- void(∗ zerokey)(u_int8_t ∗∗)

6.17.1 Detailed Description

Definition at line 48 of file xform.h.

6.17.2 Field Documentation

6.17.2.1 u_int16_t enc_xform::blocksize

Definition at line 51 of file xform.h.

Referenced by swcr_encdec().

6.17.2.2 void(∗ enc_xform::decrypt)(caddr_t, u_int8_t ∗)

Referenced by swcr_encdec().

6.17.2.3 void(∗ enc_xform::encrypt)(caddr_t, u_int8_t ∗)

Referenced by swcr_encdec().

6.17.2.4 u_int16_t enc_xform::maxkey

Definition at line 52 of file xform.h.

Referenced by cryptof_ioctl().

6.17.2.5 u_int16_t enc_xform::minkey

Definition at line 52 of file xform.h.

Referenced by cryptof_ioctl().

6.17.2.6 char* enc_xform::name

Definition at line 50 of file xform.h.

6.17.2.7 int(* enc_xform::setkey)(u_int8_t **, u_int8_t *, int len)

Referenced by swcr_encdec(), and swcr_newsession().

6.17.2.8 int enc_xform::type

Definition at line 49 of file xform.h.

Referenced by cryptof_ioctl().

6.17.2.9 void(* enc_xform::zerokey)(u_int8_t **)

Referenced by swcr_encdec(), and swcr_freesession().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/xform.h

6.18 fcrypt Struct Reference

6.18.1 Detailed Description

Definition at line 80 of file cryptodev.c.

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.c

6.19 RMD160Context Struct Reference

```
#include <rmd160.h>
```

Data Fields

- u_int32_t state [5]
- u_int64_t count
- u_char buffer [64]

6.19.1 Detailed Description

Definition at line 30 of file rmd160.h.

6.19.2 Field Documentation

6.19.2.1 u_char RMD160Context::buffer[64]

Definition at line 33 of file rmd160.h.

Referenced by RMD160Update().

6.19.2.2 u_int64_t RMD160Context::count

Definition at line 32 of file rmd160.h.

Referenced by RMD160Final(), RMD160Init(), and RMD160Update().

6.19.2.3 u_int32_t RMD160Context::state[5]

Definition at line 31 of file rmd160.h.

Referenced by RMD160Final(), RMD160Init(), and RMD160Update().

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/rmd160.h

6.20 session_op Struct Reference

```
#include <cryptodev.h>
```

Data Fields

- u_int32_t cipher
- u_int32_t mac
- u_int32_t keylen
- caddr_t key
- int mackeylen
- caddr_t mackey
- u_int32_t ses

6.20.1 Detailed Description

Definition at line 130 of file cryptodev.h.

6.20.2 Field Documentation

6.20.2.1 u_int32_t session_op::cipher

Definition at line 131 of file cryptodev.h.

Referenced by cryptof_ioctl().

6.20.2.2 caddr_t session_op::key

Definition at line 135 of file cryptodev.h.

Referenced by cryptof_ioctl().

6.20.2.3 u_int32_t session_op::keylen

Definition at line 134 of file cryptodev.h.

Referenced by cryptof_ioctl().

6.20.2.4 u_int32_t session_op::mac

Definition at line 132 of file cryptodev.h.

Referenced by cryptof_ioctl().

6.20.2.5 caddr_t session_op::mackey

Definition at line 137 of file cryptodev.h.

Referenced by cryptof_ioctl().

6.20.2.6 int session_op::mackeylen

Definition at line 136 of file cryptodev.h.

Referenced by cryptof_ioctl().

6.20.2.7 u_int32_t session_op::ses

Definition at line 139 of file cryptodev.h.

Referenced by cryptof_ioctl().

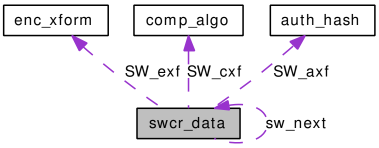
The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptodev.h

6.21 swcr_data Struct Reference

```
#include <cryptosoft.h>
```

Collaboration diagram for swcr_data:



Data Fields

- int sw_alg
- union {
  struct {
  u_int8_t ∗ SW_ictx
  u_int8_t ∗ SW_octx
  u_int16_t SW_klen
  u_int16_t SW_mlen
  auth_hash ∗ SW_axf
  } SWCR_AUTH
  struct {
  u_int8_t ∗ SW_kschedule
  enc_xform ∗ SW_exf
  } SWCR_ENC
  struct {
  u_int32_t SW_size
  comp_algo ∗ SW_cxf
  } SWCR_COMP
  } SWCR_UN
- swcr_data ∗ sw_next

6.21.1 Detailed Description

Definition at line 29 of file cryptosoft.h.

6.21.2 Field Documentation

6.21.2.1 int swcr_data::sw_alg

Definition at line 30 of file cryptosoft.h.

Referenced by swcr_authcompute(), swcr_freesession(), and swcr_process().

6.21.2.2 struct auth_hash∗ swcr_data::SW_axf

Definition at line 37 of file cryptosoft.h.

6.21.2.3 struct comp_algo∗ swcr_data::SW_cxf

Definition at line 45 of file cryptosoft.h.

6.21.2.4 struct enc_xform∗ swcr_data::SW_exf

Definition at line 41 of file cryptosoft.h.

6.21.2.5 u_int8_t∗ swcr_data::SW_ictx

Definition at line 33 of file cryptosoft.h.

6.21.2.6 u_int16_t swcr_data::SW_klen

Definition at line 35 of file cryptosoft.h.

6.21.2.7 u_int8_t∗ swcr_data::SW_kschedule

Definition at line 40 of file cryptosoft.h.

6.21.2.8 u_int16_t swcr_data::SW_mlen

Definition at line 36 of file cryptosoft.h.

6.21.2.9 struct swcr_data∗ swcr_data::sw_next

Definition at line 59 of file cryptosoft.h.

Referenced by swcr_freesession(), and swcr_process().

6.21.2.10 u_int8_t∗ swcr_data::SW_octx

Definition at line 34 of file cryptosoft.h.

6.21.2.11 u_int32_t swcr_data::SW_size

Definition at line 44 of file cryptosoft.h.

6.21.2.12 struct { ... } swcr_data::SWCR_AUTH

6.21.2.13 struct { ... } swcr_data::SWCR_COMP

6.21.2.14 struct { ... } swcr_data::SWCR_ENC

6.21.2.15 union { ... } swcr_data::SWCR_UN

The documentation for this struct was generated from the following file:

- /usr/src/sys/opencrypto/cryptosoft.h

Chapter 7

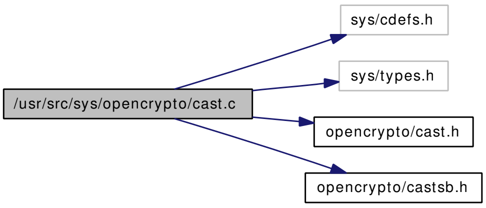
FreeBSD kernel opencrypto code File Documentation

7.1 notreviewed.dox File Reference

7.2 /usr/src/sys/opencrypto/cast.c File Reference

```
#include <sys/cdefs.h>
#include <sys/types.h>
#include <opencrypto/cast.h>
#include <opencrypto/castsb.h>
```

Include dependency graph for cast.c:



Defines

- #define U_INT8_Ta(x) ((u_int8_t) (x>>24))
- #define U_INT8_Tb(x) ((u_int8_t) ((x>>16)&255))
- #define U_INT8_Tc(x) ((u_int8_t) ((x>>8)&255))
- #define U_INT8_Td(x) ((u_int8_t) ((x)&255))
- #define ROL(x, n) (((x)<<(n)) | ((x)>>(32-(n))))
- #define F1(l, r, i)
- #define F2(l, r, i)
- #define F3(l, r, i)

Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/cast.c,v 1.3 2005/01/07 02:29:16 imp Exp $")
- void cast_encrypt (cast_key ∗key, u_int8_t ∗inblock, u_int8_t ∗outblock)
- void cast_decrypt (cast_key ∗key, u_int8_t ∗inblock, u_int8_t ∗outblock)
- void cast_setkey (cast_key ∗key, u_int8_t ∗rawkey, int keybytes)

7.2.1 Define Documentation

7.2.1.1 #define F1(l, r, i)

Value:

```
t = ROL(key->xkey[i] + r, key->xkey[i+16]); \
        l ^= ((cast_sbox1[U_INT8_Ta(t)] ^ cast_sbox2[U_INT8_Tb(t)]) - \
         cast_sbox3[U_INT8_Tc(t)]) + cast_sbox4[U_INT8_Td(t)];
```

Definition at line 26 of file cast.c.

Referenced by cast_decrypt(), cast_encrypt(), and RMD160Transform().

7.2.1.2 #define F2(l, r, i)

Value:

```
t = ROL(key->xkey[i] ^ r, key->xkey[i+16]); \
        l ^= ((cast_sbox1[U_INT8_Ta(t)] - cast_sbox2[U_INT8_Tb(t)]) + \
         cast_sbox3[U_INT8_Tc(t)]) ^ cast_sbox4[U_INT8_Td(t)];
```

Definition at line 30 of file cast.c.

Referenced by cast_decrypt(), cast_encrypt(), and RMD160Transform().

7.2.1.3 #define F3(l, r, i)

Value:

```
t = ROL(key->xkey[i] - r, key->xkey[i+16]); \
        l ^= ((cast_sbox1[U_INT8_Ta(t)] + cast_sbox2[U_INT8_Tb(t)]) ^ \
         cast_sbox3[U_INT8_Tc(t)]) - cast_sbox4[U_INT8_Td(t)];
```

Definition at line 34 of file cast.c.

Referenced by cast_decrypt(), cast_encrypt(), and RMD160Transform().

7.2.1.4 #define ROL(x, n) (((x)<<(n)) | ((x)>>(32-(n))))

Definition at line 23 of file cast.c.

7.2.1.5 #define U_INT8_Ta(x) ((u_int8_t) (x>>24))

Definition at line 17 of file cast.c.

Referenced by cast_decrypt(), cast_encrypt(), and cast_setkey().

7.2.1.6 #define U_INT8_Tb(x) ((u_int8_t) ((x>>16)&255))

Definition at line 18 of file cast.c.

Referenced by cast_decrypt(), cast_encrypt(), and cast_setkey().

7.2.1.7 #define U_INT8_Tc(x) ((u_int8_t) ((x>>8)&255))

Definition at line 19 of file cast.c.

Referenced by cast_decrypt(), cast_encrypt(), and cast_setkey().

7.2.1.8 #define U_INT8_Td(x) ((u_int8_t) ((x)&255))

Definition at line 20 of file cast.c.

Referenced by cast_decrypt(), cast_encrypt(), and cast_setkey().

7.2.2 Function Documentation

7.2.2.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/cast. *c*, v 1.3 2005/01/07 02:29:16 imp Exp $")

7.2.2.2 void cast_decrypt (cast_key ∗ *key*, u_int8_t ∗ *inblock*, u_int8_t ∗ *outblock*)

Definition at line 87 of file cast.c.

References F1, F2, F3, cast_key::rounds, U_INT8_Ta, U_INT8_Tb, U_INT8_Tc, and U_INT8_Td.

Referenced by cast5_decrypt().

7.2.2.3 void cast_encrypt (cast_key ∗ *key*, u_int8_t ∗ *inblock*, u_int8_t ∗ *outblock*)

Definition at line 42 of file cast.c.

References F1, F2, F3, cast_key::rounds, U_INT8_Ta, U_INT8_Tb, U_INT8_Tc, and U_INT8_Td.

Referenced by cast5_encrypt().

7.2.2.4 void cast_setkey (cast_key ∗ *key*, u_int8_t ∗ *rawkey*, int *keybytes*)

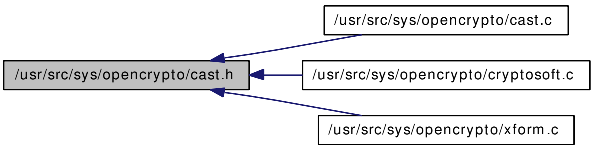
Definition at line 132 of file cast.c.

References cast_sbox5, cast_sbox6, cast_sbox7, cast_sbox8, cast_key::rounds, U_INT8_Ta, U_INT8_Tb, U_INT8_Tc, U_INT8_Td, and cast_key::xkey.

Referenced by cast5_setkey().

7.3 /usr/src/sys/opencrypto/cast.h File Reference

This graph shows which files directly or indirectly include this file:


/usr/src/sys/opencrypto/cast.h
/usr/src/sys/opencrypto/cast.c
/usr/src/sys/opencrypto/cryptosoft.c
/usr/src/sys/opencrypto/xform.c

Data Structures

- struct cast_key

Functions

- void cast_setkey (cast_key ∗key, u_int8_t ∗rawkey, int keybytes)
- void cast_encrypt (cast_key ∗key, u_int8_t ∗inblock, u_int8_t ∗outblock)
- void cast_decrypt (cast_key ∗key, u_int8_t ∗inblock, u_int8_t ∗outblock)

7.3.1 Function Documentation

7.3.1.1 void cast_decrypt (cast_key ∗ *key*, u_int8_t ∗ *inblock*, u_int8_t ∗ *outblock*)

Definition at line 87 of file cast.c.

References F1, F2, F3, cast_key::rounds, U_INT8_Ta, U_INT8_Tb, U_INT8_Tc, and U_INT8_Td.

Referenced by cast5_decrypt().

7.3.1.2 void cast_encrypt (cast_key ∗ *key*, u_int8_t ∗ *inblock*, u_int8_t ∗ *outblock*)

Definition at line 42 of file cast.c.

References F1, F2, F3, cast_key::rounds, U_INT8_Ta, U_INT8_Tb, U_INT8_Tc, and U_INT8_Td.

Referenced by cast5_encrypt().

7.3.1.3 void cast_setkey (cast_key ∗ *key*, u_int8_t ∗ *rawkey*, int *keybytes*)

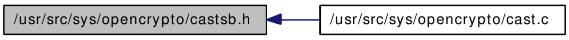
Definition at line 132 of file cast.c.

References cast_sbox5, cast_sbox6, cast_sbox7, cast_sbox8, cast_key::rounds, U_INT8_Ta, U_INT8_Tb, U_INT8_Tc, U_INT8_Td, and cast_key::xkey.

Referenced by cast5_setkey().

7.4 /usr/src/sys/opencrypto/castsb.h File Reference

This graph shows which files directly or indirectly include this file:


/usr/src/sys/opencrypto/castsb.h
/usr/src/sys/opencrypto/cast.c

Variables

- static const u_int32_t cast_sbox1 [256]
- static const u_int32_t cast_sbox2 [256]
- static const u_int32_t cast_sbox3 [256]
- static const u_int32_t cast_sbox4 [256]
- static const u_int32_t cast_sbox5 [256]
- static const u_int32_t cast_sbox6 [256]
- static const u_int32_t cast_sbox7 [256]
- static const u_int32_t cast_sbox8 [256]

7.4.1 Variable Documentation

7.4.1.1 const u_int32_t cast_sbox1[256] `[static]`

Definition at line 10 of file castsb.h.

7.4.1.2 const u_int32_t cast_sbox2[256] `[static]`

Definition at line 77 of file castsb.h.

7.4.1.3 const u_int32_t cast_sbox3[256] `[static]`

Definition at line 144 of file castsb.h.

7.4.1.4 const u_int32_t cast_sbox4[256] `[static]`

Definition at line 211 of file castsb.h.

7.4.1.5 const u_int32_t cast_sbox5[256] `[static]`

Definition at line 278 of file castsb.h.

Referenced by cast_setkey().

7.4.1.6 const u_int32_t cast_sbox6[256] `[static]`

Definition at line 345 of file castsb.h.

Referenced by cast_setkey().

7.4.1.7 const u_int32_t cast_sbox7[256] [static]

Definition at line 412 of file castsb.h.

Referenced by cast_setkey().

7.4.1.8 const u_int32_t cast_sbox8[256] [static]

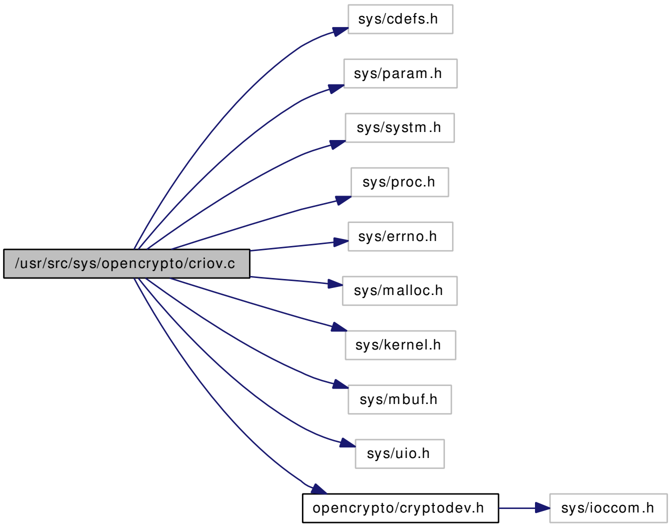
Definition at line 479 of file castsb.h.

Referenced by cast_setkey().

7.5 /usr/src/sys/opencrypto/criov.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/systm.h>
#include <sys/proc.h>
#include <sys/errno.h>
#include <sys/malloc.h>
#include <sys/kernel.h>
#include <sys/mbuf.h>
#include <sys/uio.h>
#include <opencrypto/cryptodev.h>
```

Include dependency graph for criov.c:



Defines

- #define CUIO_SKIP()

Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/criov.c,v 1.5 2006/06/04 22:15:13 pjd Exp $")
- void cuio_copydata (struct uio ∗uio, int off, int len, caddr_t cp)
- void cuio_copyback (struct uio ∗uio, int off, int len, caddr_t cp)
- iovec ∗ cuio_getptr (struct uio ∗uio, int loc, int ∗off)
- int cuio_apply (struct uio ∗uio, int off, int len, int(∗f)(void ∗, void ∗, u_int), void ∗arg)
- void crypto_copyback (int flags, caddr_t buf, int off, int size, caddr_t in)
- void crypto_copydata (int flags, caddr_t buf, int off, int size, caddr_t out)
- int crypto_apply (int flags, caddr_t buf, int off, int len, int(∗f)(void ∗, void ∗, u_int), void ∗arg)

7.5.1 Define Documentation

7.5.1.1 #define CUIO_SKIP()

Value:

```
do {                                                        \
        KASSERT(off >= 0, ("%s: off %d < 0", __func__, off));          \
        KASSERT(len >= 0, ("%s: len %d < 0", __func__, len));          \
        while (off > 0) {                                              \
                KASSERT(iol >= 0, ("%s: empty in skip", __func__));    \
                if (off < iov->iov_len)                                \
                        break;                                         \
                off -= iov->iov_len;                                   \
                iol--;                                                 \
                iov++;                                                 \
        }                                                              \
} while (0)
```

Definition at line 48 of file criov.c.

Referenced by cuio_apply(), cuio_copyback(), and cuio_copydata().

7.5.2 Function Documentation

7.5.2.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/criov. *c*, v 1.5 2006/06/04 22:15:13 pjd Exp $")

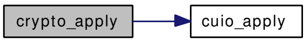
7.5.2.2 int crypto_apply (int *flags*, caddr_t *buf*, int *off*, int *len*, int(∗)(void ∗, void ∗, u_int) *f*, void ∗ *arg*)

Definition at line 186 of file criov.c.

References CRYPTO_F_IMBUF, CRYPTO_F_IOV, and cuio_apply().

Referenced by swcr_authcompute().

Here is the call graph for this function:



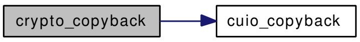
7.5.2.3 void crypto_copyback (int *flags*, caddr_t *buf*, int *off*, int *size*, caddr_t *in*)

Definition at line 162 of file criov.c.

References CRYPTO_F_IMBUF, CRYPTO_F_IOV, and cuio_copyback().

Referenced by swcr_authcompute(), swcr_compdec(), and swcr_encdec().

Here is the call graph for this function:



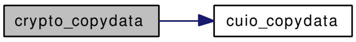
7.5.2.4 void crypto_copydata (int *flags*, caddr_t *buf*, int *off*, int *size*, caddr_t *out*)

Definition at line 174 of file criov.c.

References CRYPTO_F_IMBUF, CRYPTO_F_IOV, and cuio_copydata().

Referenced by swcr_compdec(), and swcr_encdec().

Here is the call graph for this function:



7.5.2.5 int cuio_apply (struct uio ∗ *uio*, int *off*, int *len*, int(∗)(void ∗, void ∗, u_int) *f*, void ∗ *arg*)

Definition at line 138 of file criov.c.

References CUIO_SKIP.

Referenced by crypto_apply().

7.5.2.6 void cuio_copyback (struct uio ∗ *uio*, int *off*, int *len*, caddr_t *cp*)

Definition at line 82 of file criov.c.

References CUIO_SKIP.

Referenced by crypto_copyback(), and swcr_encdec().

7.5.2.7 void cuio_copydata (struct uio ∗ *uio*, int *off*, int *len*, caddr_t *cp*)

Definition at line 62 of file criov.c.

References CUIO_SKIP.

Referenced by crypto_copydata(), and swcr_encdec().

7.5.2.8 struct iovec∗ cuio_getptr (struct uio ∗ *uio*, int *loc*, int ∗ *off*)

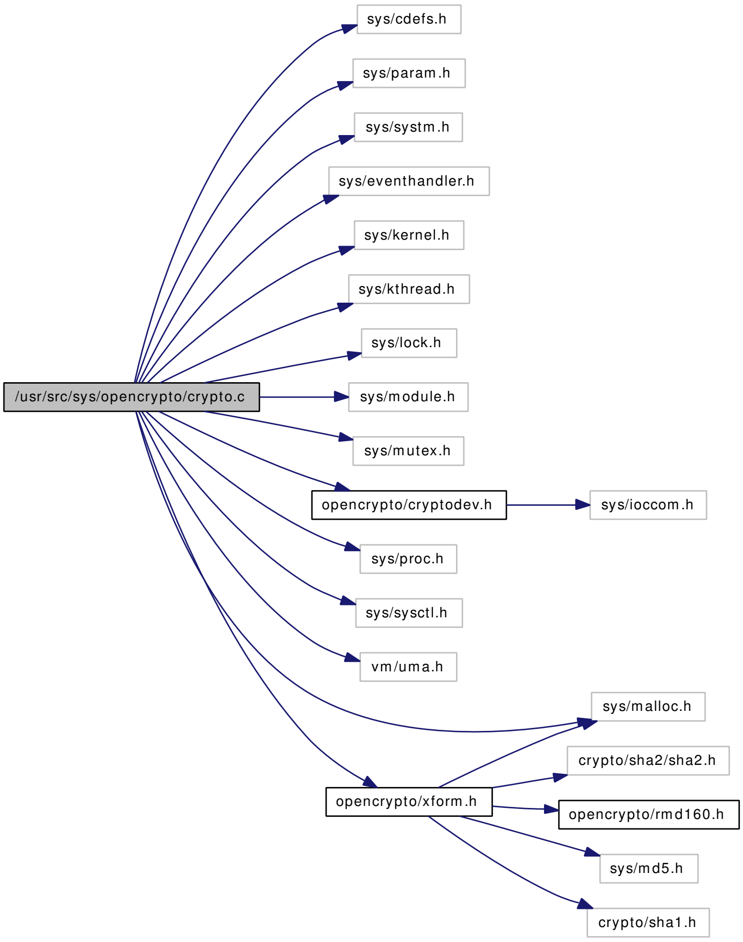
Definition at line 105 of file criov.c.

Referenced by swcr_encdec().

7.6 /usr/src/sys/opencrypto/crypto.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/systm.h>
#include <sys/eventhandler.h>
#include <sys/kernel.h>
#include <sys/kthread.h>
#include <sys/lock.h>
#include <sys/module.h>
#include <sys/mutex.h>
#include <sys/malloc.h>
#include <sys/proc.h>
#include <sys/sysctl.h>
#include <vm/uma.h>
#include <opencrypto/cryptodev.h>
#include <opencrypto/xform.h>
```

Include dependency graph for crypto.c:



Defines

- #define CRYPTO_TIMING
- #define CRYPTO_DRIVER_LOCK() mtx_lock(&crypto_drivers_mtx)
- #define CRYPTO_DRIVER_UNLOCK() mtx_unlock(&crypto_drivers_mtx)
- #define CRYPTO_Q_LOCK() mtx_lock(&crypto_q_mtx)
- #define CRYPTO_Q_UNLOCK() mtx_unlock(&crypto_q_mtx)
- #define CRYPTO_RETQ_LOCK() mtx_lock(&crypto_ret_q_mtx)
- #define CRYPTO_RETQ_UNLOCK() mtx_unlock(&crypto_ret_q_mtx)
- #define CRYPTO_RETQ_EMPTY() (TAILQ_EMPTY(&crp_ret_q) && TAILQ_EMPTY(&crp_ret_kq))

Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/crypto.c,v 1.26 2006/06/06 15:04:52 pjd Exp $")
- static TAILQ_HEAD (cryptop)
- static void crypto_terminate (struct proc ∗∗pp, void ∗q)
- static void crypto_destroy (void)
- static int crypto_modevent (module_t mod, int type, void ∗unused)
- MODULE_VERSION (crypto, 1)

- DECLARE_MODULE (crypto, crypto_mod, SI_SUB_DRIVERS, SI_ORDER_FIRST)
- MODULE_DEPEND (crypto, zlib, 1, 1, 1)
- int crypto_newsession (u_int64_t ∗sid, struct cryptoini ∗cri, int hard)
- static void crypto_remove (struct cryptocap ∗cap)
- int crypto_freesession (u_int64_t sid)
- int32_t crypto_get_driverid (u_int32_t flags)
- static struct cryptocap ∗ crypto_checkdriver (u_int32_t hid)
- int crypto_kregister (u_int32_t driverid, int kalg, u_int32_t flags, int(∗kprocess)(void ∗, struct cryptkop ∗, int), void ∗karg)
- int crypto_register (u_int32_t driverid, int alg, u_int16_t maxoplen, u_int32_t flags, int(∗newses)(void ∗, u_int32_t ∗, struct cryptoini ∗), int(∗freeses)(void ∗, u_int64_t), int(∗process)(void ∗, struct cryptop ∗, int), void ∗arg)
- int crypto_unregister (u_int32_t driverid, int alg)
- int crypto_unregister_all (u_int32_t driverid)
- int crypto_unblock (u_int32_t driverid, int what)
- int crypto_dispatch (struct cryptop ∗crp)
- int crypto_kdispatch (struct cryptkop ∗krp)
- static int crypto_kinvoke (struct cryptkop ∗krp)
- static void crypto_tstat (struct cryptotstat ∗ts, struct bintime ∗bt)
- static int crypto_invoke (struct cryptocap ∗cap, struct cryptop ∗crp, int hint)
- void crypto_freereq (struct cryptop ∗crp)
- cryptop ∗ crypto_getreq (int num)
- void crypto_done (struct cryptop ∗crp)
- void crypto_kdone (struct cryptkop ∗krp)
- int crypto_getfeat (int ∗featp)
- static void crypto_finis (void ∗chan)
- static void crypto_proc (void)
- static void crypto_ret_proc (void)

Variables

- static struct mtx crypto_drivers_mtx
- static struct cryptocap ∗ crypto_drivers = NULL
- static int crypto_drivers_num = 0
- static int crp_sleep = 0
- static moduledata_t crypto_mod

7.6.1 Define Documentation

7.6.1.1 #define CRYPTO_DRIVER_LOCK() mtx_lock(&crypto_drivers_mtx)

Definition at line 50 of file crypto.c.

Referenced by crypto_destroy(), crypto_finis(), crypto_freesession(), crypto_get_driverid(), crypto_getfeat(), crypto_kdone(), crypto_kinvoke(), crypto_kregister(), crypto_newsession(), crypto_register(), crypto_terminate(), crypto_unregister(), and crypto_unregister_all().

7.6.1.2 #define CRYPTO_DRIVER_UNLOCK() mtx_unlock(&crypto_drivers_mtx)

Definition at line 51 of file crypto.c.

Referenced by crypto_destroy(), crypto_finis(), crypto_freesession(), crypto_get_driverid(), crypto_getfeat(), crypto_kdone(), crypto_kregister(), crypto_newsession(), crypto_register(), crypto_terminate(), crypto_unregister(), and crypto_unregister_all().

7.6.1.3 #define CRYPTO_Q_LOCK() mtx_lock(&crypto_q_mtx)

Referenced by crypto_dispatch(), crypto_freereq(), crypto_kdispatch(), crypto_proc(), and crypto_unblock().

7.6.1.4 #define CRYPTO_Q_UNLOCK() mtx_unlock(&crypto_q_mtx)

Referenced by crypto_dispatch(), crypto_freereq(), crypto_kdispatch(), crypto_proc(), and crypto_unblock().

7.6.1.5 #define CRYPTO_RETQ_EMPTY() (TAILQ_EMPTY(&crp_ret_q) && TAILQ_EMPTY(&crp_ret_kq))

Referenced by crypto_done(), and crypto_kdone().

7.6.1.6 #define CRYPTO_RETQ_LOCK() mtx_lock(&crypto_ret_q_mtx)

Referenced by crypto_done(), crypto_freereq(), crypto_kdone(), and crypto_ret_proc().

7.6.1.7 #define CRYPTO_RETQ_UNLOCK() mtx_unlock(&crypto_ret_q_mtx)

Referenced by crypto_done(), crypto_freereq(), crypto_kdone(), and crypto_ret_proc().

7.6.1.8 #define CRYPTO_TIMING

Definition at line 26 of file crypto.c.

7.6.2 Function Documentation

7.6.2.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/crypto. *c*, v 1.26 2006/06/06 15:04:52 pjd Exp $")

7.6.2.2 static struct cryptocap∗ crypto_checkdriver (u_int32_t *hid*) [static]

Definition at line 479 of file crypto.c.

References crypto_drivers, and crypto_drivers_num.

Referenced by crypto_dispatch(), crypto_kregister(), crypto_proc(), crypto_register(), crypto_unblock(), crypto_unregister(), and crypto_unregister_all().

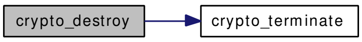
7.6.2.3 static void crypto_destroy (void) [static]

Definition at line 200 of file crypto.c.

References CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_mtx, and crypto_terminate().

Referenced by crypto_modevent().

Here is the call graph for this function:



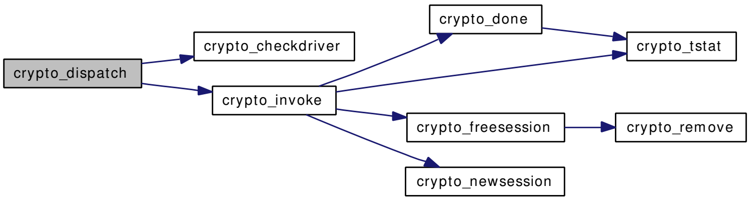
7.6.2.4 int crypto_dispatch (struct cryptop ∗ *crp*)

Definition at line 701 of file crypto.c.

References cryptocap::cc_qblocked, cryptop::crp_flags, cryptop::crp_sid, crp_sleep, cryptop::crp_tstamp, crypto_checkdriver(), CRYPTO_F_BATCH, crypto_invoke(), CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, CRYPTO_SESID2HID, and cryptostats::cs_ops.

Referenced by cryptodev_cb(), and cryptodev_op().

Here is the call graph for this function:



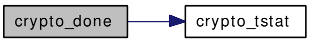
7.6.2.5 void crypto_done (struct cryptop ∗ *crp*)

Definition at line 964 of file crypto.c.

References cryptop::crp_callback, cryptop::crp_etype, cryptop::crp_flags, cryptop::crp_sid, cryptop::crp_tstamp, CRYPTO_F_CBIFSYNC, CRYPTO_F_CBIMM, CRYPTO_F_DONE, CRYPTO_RETQ_EMPTY, CRYPTO_RETQ_LOCK, CRYPTO_RETQ_UNLOCK, CRYPTO_SESID2CAPS, crypto_tstat(), CRYPTOCAP_F_SYNC, cryptostats::cs_cb, cryptostats::cs_done, cryptostats::cs_errs, and cryptostats::cs_finis.

Referenced by crypto_invoke().

Here is the call graph for this function:



7.6.2.6 static void crypto_finis (void ∗ *chan*) [static]

Definition at line 1080 of file crypto.c.

References CRYPTO_DRIVER_LOCK, and CRYPTO_DRIVER_UNLOCK.

Referenced by crypto_proc(), and crypto_ret_proc().

7.6.2.7 void crypto_freereq (struct cryptop ∗ *crp*)

Definition at line 899 of file crypto.c.

References cryptodesc::crd_next, cryptop::crp_desc, CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, CRYPTO_RETQ_LOCK, and CRYPTO_RETQ_UNLOCK.

Referenced by crypto_getreq(), and cryptodev_op().

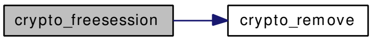
7.6.2.8 int crypto_freesession (u_int64_t *sid*)

Definition at line 383 of file crypto.c.

References cryptocap::cc_arg, cryptocap::cc_flags, cryptocap::cc_freesession, cryptocap::cc_sessions, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, crypto_remove(), CRYPTO_SESID2HID, and CRYPTOCAP_F_CLEANUP.

Referenced by crypto_invoke(), cryptof_ioctl(), and csefree().

Here is the call graph for this function:



7.6.2.9 int32_t crypto_get_driverid (u_int32_t *flags*)

Definition at line 427 of file crypto.c.

References cryptocap::cc_flags, cryptocap::cc_process, cryptocap::cc_sessions, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, and CRYPTOCAP_F_CLEANUP.

Referenced by swcr_init().

7.6.2.10 int crypto_getfeat (int ∗ *featp*)

Definition at line 1044 of file crypto.c.

References cryptocap::cc_flags, cryptocap::cc_kalg, cryptocap::cc_kprocess, CRK_ALGORITHM_MAX, CRYPTO_ALG_FLAG_SUPPORTED, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, and CRYPTOCAP_F_SOFTWARE.

Referenced by cryptof_ioctl().

7.6.2.11 struct cryptop∗ crypto_getreq (int *num*)

Definition at line 939 of file crypto.c.

References cryptodesc::crd_next, cryptop::crp_desc, and crypto_freereq().

Referenced by cryptodev_op().

Here is the call graph for this function:

crypto_getreq
crypto_freereq

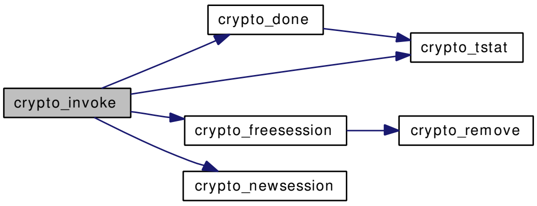
7.6.2.12 static int crypto_invoke (struct cryptocap * *cap*, struct cryptop * *crp*, int *hint*) [static]

Definition at line 853 of file crypto.c.

References cryptocap::cc_arg, cryptocap::cc_flags, cryptocap::cc_process, cryptodesc::CRD_INI, cryptodesc::crd_next, cryptoini::cri_next, cryptop::crp_callback, cryptop::crp_desc, cryptop::crp_etype, cryptop::crp_sid, cryptop::crp_tstamp, crypto_done(), crypto_freesession(), crypto_newsession(), crypto_tstat(), CRYPTOCAP_F_CLEANUP, and cryptostats::cs_invoke.

Referenced by crypto_dispatch(), and crypto_proc().

Here is the call graph for this function:



7.6.2.13 int crypto_kdispatch (struct cryptkop * *krp*)

Definition at line 748 of file crypto.c.

References crp_sleep, crypto_kinvoke(), CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, and cryptostats::cs_kops.

Referenced by cryptodev_key().

Here is the call graph for this function:

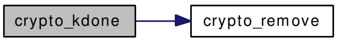
crypto_kdispatch
crypto_kinvoke

7.6.2.14 void crypto_kdone (struct cryptkop * *krp*)

Definition at line 1020 of file crypto.c.

References cryptocap::cc_flags, cryptocap::cc_koperations, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, crypto_remove(), CRYPTO_RETQ_EMPTY, CRYPTO_RETQ_LOCK, CRYPTO_RETQ_UNLOCK, CRYPTOCAP_F_CLEANUP, cryptostats::cs_-kerrs, cryptkop::krp_hid, and cryptkop::krp_status.

Here is the call graph for this function:



7.6.2.15 static int crypto_kinvoke (struct cryptkop ∗ *krp*) [static]

Definition at line 770 of file crypto.c.

References cryptocap::cc_flags, cryptocap::cc_kalg, cryptocap::cc_kprocess, cryptocap::cc_kqblocked, CRYPTO_ALG_FLAG_SUPPORTED, CRYPTO_DRIVER_LOCK, crypto_drivers, crypto_drivers_num, CRYPTOCAP_F_SOFTWARE, cryptkop::krp_callback, and cryptkop::krp_op.

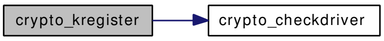
Referenced by crypto_kdispatch(), and crypto_proc().

7.6.2.16 int crypto_kregister (u_int32_t *driverid*, int *kalg*, u_int32_t *flags*, int(∗)(void ∗, struct cryptkop ∗, int) *kprocess*, void ∗ *karg*)

Definition at line 491 of file crypto.c.

References cryptocap::cc_kalg, cryptocap::cc_karg, cryptocap::cc_kprocess, CRK_ALGORITHM_MAX, CRK_ALGORITM_MIN, CRYPTO_ALG_FLAG_SUPPORTED, crypto_checkdriver(), CRYPTO_-DRIVER_LOCK, and CRYPTO_DRIVER_UNLOCK.

Here is the call graph for this function:

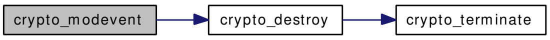


7.6.2.17 static int crypto_modevent (module_t *mod*, int *type*, void ∗ *unused*) [static]

Definition at line 231 of file crypto.c.

References crypto_destroy().

Here is the call graph for this function:



7.6.2.18 int crypto_newsession (u_int64_t ∗ *sid*, struct cryptoini ∗ *cri*, int *hard*)

Definition at line 263 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_arg, cryptocap::cc_flags, cryptocap::cc_newsession, cryptocap::cc_sessions, cryptoini::cri_alg, cryptoini::cri_next, CRYPTO_DRIVER_LOCK, CRYPTO_-DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, CRYPTOCAP_F_CLEANUP, and CRYPTOCAP_F_SOFTWARE.

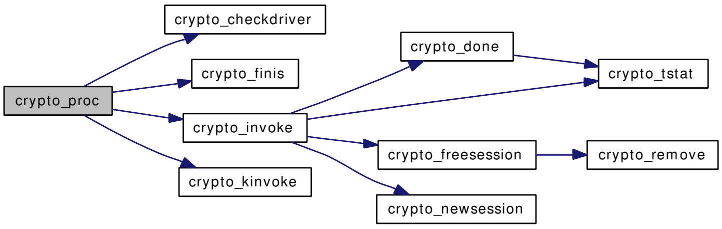
Referenced by crypto_invoke(), and cryptof_ioctl().

7.6.2.19 static void crypto_proc (void) [static]

Definition at line 1092 of file crypto.c.

References cryptocap::cc_kprocess, cryptocap::cc_kqblocked, cryptocap::cc_process, cryptocap::cc_qblocked, cryptop::crp_flags, cryptop::crp_sid, crp_sleep, crypto_checkdriver(), crypto_drivers, CRYPTO_F_BATCH, crypto_finis(), CRYPTO_HINT_MORE, crypto_invoke(), crypto_kinvoke(), CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, CRYPTO_SESID2HID, cryptostats::cs_blocks, cryptostats::cs_intrs, cryptostats::cs_kblocks, and cryptkop::krp_hid.

Here is the call graph for this function:



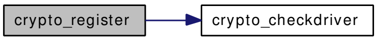
7.6.2.20 int crypto_register (u_int32_t *driverid*, int *alg*, u_int16_t *maxoplen*, u_int32_t *flags*, int(∗)(void ∗, u_int32_t ∗, struct cryptoini ∗) *newses*, int(∗)(void ∗, u_int64_t) *freeses*, int(∗)(void ∗, struct cryptop ∗, int) *process*, void ∗ *arg*)

Definition at line 534 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_arg, cryptocap::cc_freesession, cryptocap::cc_max_op_len, cryptocap::cc_newsession, cryptocap::cc_process, cryptocap::cc_sessions, CRYPTO_ALG_FLAG_SUPPORTED, CRYPTO_ALGORITHM_MAX, CRYPTO_ALGORITHM_MIN, crypto_checkdriver(), CRYPTO_DRIVER_LOCK, and CRYPTO_DRIVER_UNLOCK.

Referenced by swcr_init().

Here is the call graph for this function:



7.6.2.21 static void crypto_remove (struct cryptocap ∗ *cap*) [static]

Definition at line 370 of file crypto.c.

References cryptocap::cc_koperations, cryptocap::cc_sessions, and crypto_drivers_mtx.

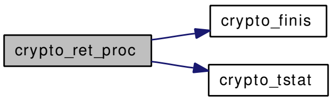
Referenced by crypto_freesession(), and crypto_kdone().

7.6.2.22 static void crypto_ret_proc (void) [static]

Definition at line 1231 of file crypto.c.

References cryptop::crp_callback, cryptop::crp_tstamp, crypto_finis(), CRYPTO_RETQ_LOCK, CRYPTO_RETQ_UNLOCK, crypto_tstat(), cryptostats::cs_cb, cryptostats::cs_finis, cryptostats::cs_rets, and cryptkop::krp_callback.

Here is the call graph for this function:



7.6.2.23 static void crypto_terminate (struct proc ∗∗ *pp*, void ∗ *q*) [static]

Definition at line 182 of file crypto.c.

References CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, and crypto_drivers_mtx.

Referenced by crypto_destroy().

7.6.2.24 static void crypto_tstat (struct cryptotstat ∗ *ts*, struct bintime ∗ *bt*) [static]

Definition at line 825 of file crypto.c.

References cryptotstat::acc, cryptotstat::count, cryptotstat::max, and cryptotstat::min.

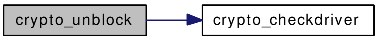
Referenced by crypto_done(), crypto_invoke(), and crypto_ret_proc().

7.6.2.25 int crypto_unblock (u_int32_t *driverid*, int *what*)

Definition at line 675 of file crypto.c.

References cryptocap::cc_kqblocked, cryptocap::cc_qblocked, crp_sleep, CRYPTO_ASYMQ, crypto_checkdriver(), CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, and CRYPTO_SYMQ.

Here is the call graph for this function:

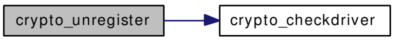


7.6.2.26 int crypto_unregister (u_int32_t *driverid*, int *alg*)

Definition at line 588 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_koperations, cryptocap::cc_max_op_len, cryptocap::cc_sessions, CRYPTO_ALGORITHM_MAX, CRYPTO_ALGORITHM_MIN, crypto_checkdriver(), CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, and CRYPTOCAP_F_CLEANUP.

Here is the call graph for this function:

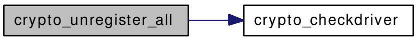


7.6.2.27 int crypto_unregister_all (u_int32_t *driverid*)

Definition at line 637 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_koperations, cryptocap::cc_max_op_len, cryptocap::cc_sessions, CRYPTO_ALGORITHM_MAX, CRYPTO_ALGORITHM_MIN, crypto_checkdriver(), CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, and CRYPTOCAP_F_CLEANUP.

Here is the call graph for this function:



7.6.2.28 DECLARE_MODULE (crypto, crypto_mod, SI_SUB_DRIVERS, SI_ORDER_FIRST)

7.6.2.29 MODULE_DEPEND (crypto, zlib, 1, 1, 1)

7.6.2.30 MODULE_VERSION (crypto, 1)

7.6.2.31 static TAILQ_HEAD (cryptop) `[static]`

Definition at line 63 of file crypto.c.

References crypto_devallowsoft, and crypto_userasymcrypto.

7.6.3 Variable Documentation

7.6.3.1 int crp_sleep = 0 `[static]`

Definition at line 62 of file crypto.c.

Referenced by crypto_dispatch(), crypto_kdispatch(), crypto_proc(), and crypto_unblock().

7.6.3.2 struct cryptocap* crypto_drivers = NULL `[static]`

Definition at line 52 of file crypto.c.

Referenced by crypto_checkdriver(), crypto_destroy(), crypto_freesession(), crypto_get_driverid(), crypto_getfeat(), crypto_kdone(), crypto_kinvoke(), crypto_newsession(), and crypto_proc().

7.6.3.3 struct mtx crypto_drivers_mtx `[static]`

Definition at line 49 of file crypto.c.

Referenced by crypto_destroy(), crypto_remove(), and crypto_terminate().

7.6.3.4 int crypto_drivers_num = 0 `[static]`

Definition at line 53 of file crypto.c.

Referenced by crypto_checkdriver(), crypto_freesession(), crypto_get_driverid(), crypto_getfeat(), crypto_kdone(), crypto_kinvoke(), and crypto_newsession().

7.6.3.5 moduledata_t crypto_mod [static]

Initial value:

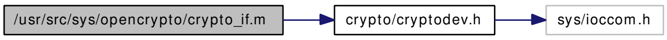
```
 {
        "crypto",
        crypto_modevent,
        0
}
```

Definition at line 250 of file crypto.c.

7.7 /usr/src/sys/opencrypto/crypto_if.m File Reference

```
#include <crypto/cryptodev.h>
```

Include dependency graph for crypto_if.m:



Variables

- INTERFACE crypto

7.7.1 Variable Documentation

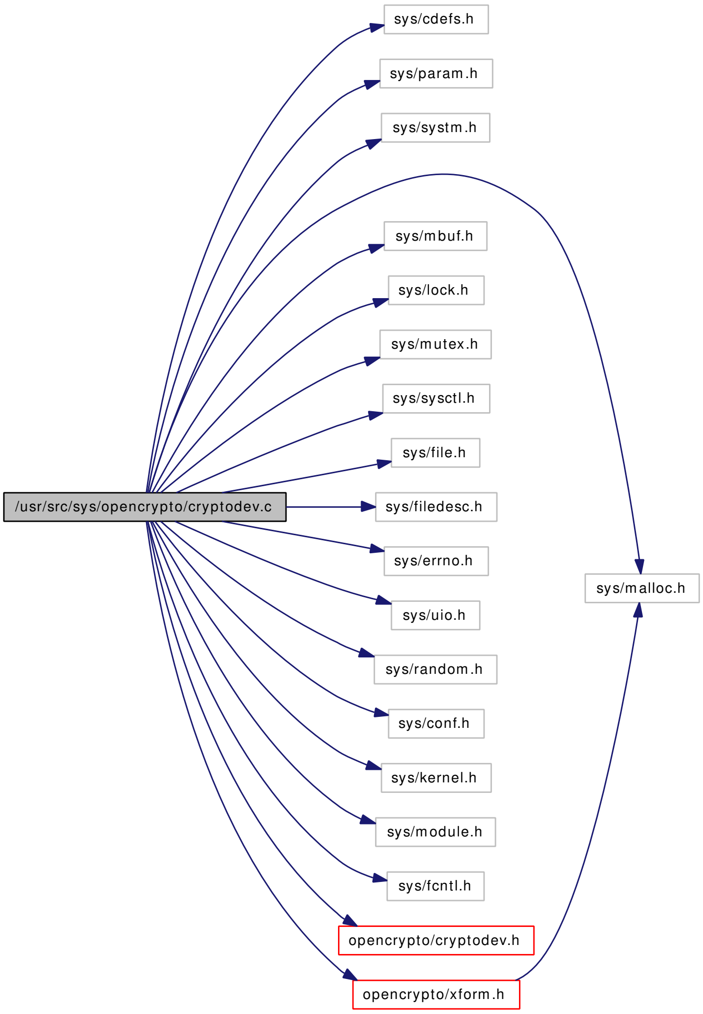
7.7.1.1 INTERFACE crypto

Definition at line 37 of file crypto_if.m.

7.8 /usr/src/sys/opencrypto/cryptodev.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/systm.h>
#include <sys/malloc.h>
#include <sys/mbuf.h>
#include <sys/lock.h>
#include <sys/mutex.h>
#include <sys/sysctl.h>
#include <sys/file.h>
#include <sys/filedesc.h>
#include <sys/errno.h>
#include <sys/uio.h>
#include <sys/random.h>
#include <sys/conf.h>
#include <sys/kernel.h>
#include <sys/module.h>
#include <sys/fcntl.h>
#include <opencrypto/cryptodev.h>
#include <opencrypto/xform.h>
```

Include dependency graph for cryptodev.c:



Data Structures

- struct csession
- struct fcrypt

Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/cryptodev.c,v 1.31 2006/05/22 16:24:11 pjd Exp $")
- static int cryptof_rw (struct file ∗fp, struct uio ∗uio, struct ucred ∗cred, int flags, struct thread ∗)
- static int cryptof_ioctl (struct file ∗, u_long, void ∗, struct ucred ∗, struct thread ∗)
- static int cryptof_poll (struct file ∗, int, struct ucred ∗, struct thread ∗)
- static int cryptof_kqfilter (struct file ∗, struct knote ∗)
- static int cryptof_stat (struct file ∗, struct stat ∗, struct ucred ∗, struct thread ∗)
- static int cryptof_close (struct file ∗, struct thread ∗)
- static struct csession ∗ csefind (struct fcrypt ∗, u_int)
- static int csedelete (struct fcrypt ∗, struct csession ∗)
- static struct csession ∗ cseadd (struct fcrypt ∗, struct csession ∗)

- static struct csession ∗ csecreate (struct fcrypt ∗, u_int64_t, caddr_t, u_int64_t, caddr_t, u_int64_t, u_int32_t, u_int32_t, struct enc_xform ∗, struct auth_hash ∗)
- static int csefree (struct csession ∗)
- static int cryptodev_op (struct csession ∗, struct crypt_op ∗, struct ucred ∗, struct thread ∗td)
- static int cryptodev_key (struct crypt_kop ∗)
- static int cryptodev_cb (void ∗)
- static int cryptodevkey_cb (void ∗op)
- static int cryptoopen (struct cdev ∗dev, int oflags, int devtype, struct thread ∗td)
- static int cryptoread (struct cdev ∗dev, struct uio ∗uio, int ioflag)
- static int cryptowrite (struct cdev ∗dev, struct uio ∗uio, int ioflag)
- static int cryptoioctl (struct cdev ∗dev, u_long cmd, caddr_t data, int flag, struct thread ∗td)
- static int cryptodev_modevent (module_t mod, int type, void ∗unused)
- MODULE_VERSION (cryptodev, 1)
- DECLARE_MODULE (cryptodev, cryptodev_mod, SI_SUB_PSEUDO, SI_ORDER_ANY)
- MODULE_DEPEND (cryptodev, crypto, 1, 1, 1)
- MODULE_DEPEND (cryptodev, zlib, 1, 1, 1)

Variables

- static struct fileops cryptofops
- static struct cdevsw crypto_cdevsw
- static struct cdev ∗ crypto_dev
- static moduledata_t cryptodev_mod

7.8.1 Function Documentation

7.8.1.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/cryptodev. *c*, v 1.31 2006/05/22 16:24:11 pjd Exp $")

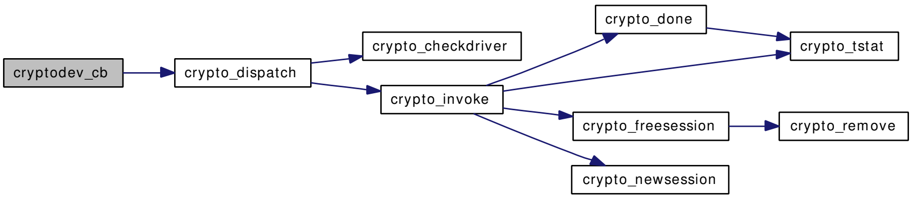
7.8.1.2 static int cryptodev_cb (void ∗) `[static]`

Definition at line 484 of file cryptodev.c.

References cryptop::crp_etype, cryptop::crp_opaque, and crypto_dispatch().

Referenced by cryptodev_op().

Here is the call graph for this function:



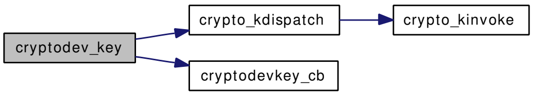
7.8.1.3 static int cryptodev_key (struct crypt_kop ∗) [static]

Definition at line 508 of file cryptodev.c.

References CRK_DH_COMPUTE_KEY, CRK_DSA_SIGN, CRK_DSA_VERIFY, crypt_kop::crk_iparams, CRK_MAXPARAM, CRK_MOD_EXP, CRK_MOD_EXP_CRT, crypt_kop::crk_op, crypt_kop::crk_oparams, crypt_kop::crk_param, crypt_kop::crk_status, crparam::crp_nbits, crparam::crp_p, crypto_kdispatch(), cryptodevkey_cb(), and cryptkop::krp_iparams.

Referenced by cryptof_ioctl().

Here is the call graph for this function:



7.8.1.4 static int cryptodev_modevent (module_t *mod*, int *type*, void ∗ *unused*) [static]

Definition at line 800 of file cryptodev.c.

References crypto_cdevsw, and crypto_dev.

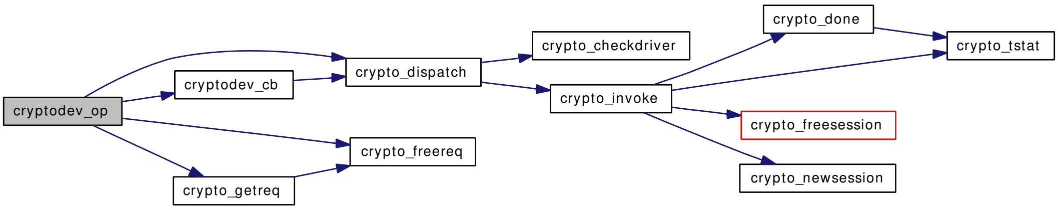
7.8.1.5 static int cryptodev_op (struct csession ∗, struct crypt_op ∗, struct ucred ∗, struct thread ∗ *td*) [static]

Definition at line 328 of file cryptodev.c.

References COP_ENCRYPT, COP_F_BATCH, CRD_F_ENCRYPT, CRD_F_IV_EXPLICIT, CRD_F_IV_PRESENT, cryptodesc::crd_flags, cryptodesc::crd_inject, cryptodesc::crd_len, cryptodesc::crd_next, cryptodesc::crd_skip, cryptop::crp_buf, cryptop::crp_callback, cryptop::crp_desc, cryptop::crp_etype, cryptop::crp_flags, cryptop::crp_ilen, cryptop::crp_opaque, cryptop::crp_sid, CRYPTO_ARC4, crypto_dispatch(), CRYPTO_F_CBIMM, CRYPTO_F_DONE, CRYPTO_F_IOV, crypto_freereq(), crypto_getreq(), cryptodev_cb(), crypt_op::dst, crypt_op::flags, crypt_op::iv, crypt_op::len, crypt_op::mac, crypt_op::op, and crypt_op::src.

Referenced by cryptof_ioctl().

Here is the call graph for this function:



7.8.1.6 static int cryptodevkey_cb (void ∗ *op*) [static]

Definition at line 499 of file cryptodev.c.

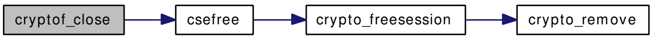
Referenced by cryptodev_key().

7.8.1.7 static int cryptof_close (struct file ∗, struct thread ∗) [static]

Definition at line 639 of file cryptodev.c.

References csefree().

Here is the call graph for this function:

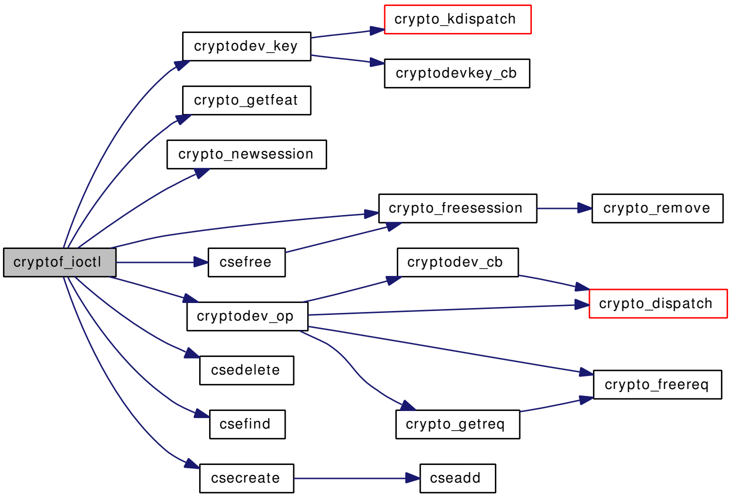


7.8.1.8 static int cryptof_ioctl (struct file ∗, u_long, void ∗, struct ucred ∗, struct thread ∗) [static]

Definition at line 131 of file cryptodev.c.

References auth_hash_hmac_md5, auth_hash_hmac_ripemd_160, auth_hash_hmac_sha1, auth_hash_hmac_sha2_256, auth_hash_hmac_sha2_384, auth_hash_hmac_sha2_512, auth_hash_null, CIOCASYMFEAT, CIOCCRYPT, CIOCFSESSION, CIOCGSESSION, CIOCKEY, session_op::cipher, CRYPTO_3DES_CBC, CRYPTO_AES_CBC, CRYPTO_ARC4, CRYPTO_BLF_CBC, CRYPTO_CAST_CBC, CRYPTO_DES_CBC, crypto_devallowsoft, crypto_freesession(), crypto_getfeat(), CRYPTO_MD5, CRYPTO_MD5_HMAC, crypto_newsession(), CRYPTO_NULL_CBC, CRYPTO_NULL_HMAC, CRYPTO_RIPEMD160_HMAC, CRYPTO_SHA1, CRYPTO_SHA1_HMAC, CRYPTO_SHA2_256_HMAC, CRYPTO_SHA2_384_HMAC, CRYPTO_SHA2_512_HMAC, CRYPTO_SKIPJACK_CBC, cryptodev_key(), cryptodev_op(), csecreate(), csedelete(), csefind(), csefree(), enc_xform_3des, enc_xform_arc4, enc_xform_blf, enc_xform_cast5, enc_xform_des, enc_xform_null, enc_xform_rijndael128, enc_xform_skipjack, session_op::key, session_op::keylen, auth_hash::keysize, session_op::mac, session_op::mackey, session_op::mackeylen, enc_xform::maxkey, enc_xform::minkey, crypt_op::ses, session_op::ses, auth_hash::type, and enc_xform::type.

Here is the call graph for this function:



7.8.1.9 static int cryptof_kqfilter (struct file ∗, struct knote ∗) [static]

Definition at line 619 of file cryptodev.c.

7.8.1.10 static int cryptof_poll (struct file ∗, int, struct ucred ∗, struct thread ∗) `[static]`

Definition at line 607 of file cryptodev.c.

7.8.1.11 static int cryptof_rw (struct file ∗ *fp*, struct uio ∗ *uio*, struct ucred ∗ *cred*, int *flags*, struct thread ∗) `[static]`

Definition at line 118 of file cryptodev.c.

7.8.1.12 static int cryptof_stat (struct file ∗, struct stat ∗, struct ucred ∗, struct thread ∗) `[static]`

Definition at line 627 of file cryptodev.c.

7.8.1.13 static int cryptoioctl (struct cdev ∗ *dev*, u_long *cmd*, caddr_t *data*, int *flag*, struct thread ∗ *td*) `[static]`

Definition at line 751 of file cryptodev.c.

References CRIOGET, and cryptofops.

7.8.1.14 static int cryptoopen (struct cdev ∗ *dev*, int *oflags*, int *devtype*, struct thread ∗ *td*) `[static]`

Definition at line 733 of file cryptodev.c.

7.8.1.15 static int cryptoread (struct cdev ∗ *dev*, struct uio ∗ *uio*, int *ioflag*) `[static]`

Definition at line 739 of file cryptodev.c.

7.8.1.16 static int cryptowrite (struct cdev ∗ *dev*, struct uio ∗ *uio*, int *ioflag*) `[static]`

Definition at line 745 of file cryptodev.c.

7.8.1.17 static struct csession ∗ cseadd (struct fcrypt ∗, struct csession ∗) `[static]`

Definition at line 679 of file cryptodev.c.

Referenced by csecreate().

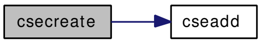
7.8.1.18 struct csession ∗ csecreate (struct fcrypt ∗, u_int64_t, caddr_t, u_int64_t, caddr_t, u_int64_t, u_int32_t, u_int32_t, struct enc_xform ∗, struct auth_hash ∗) `[static]`

Definition at line 687 of file cryptodev.c.

References cseadd().

Referenced by cryptof_ioctl().

Here is the call graph for this function:



7.8.1.19 static int csedelete (struct fcrypt ∗, struct csession ∗) [static]

Definition at line 665 of file cryptodev.c.

Referenced by cryptof_ioctl().

7.8.1.20 static struct csession ∗ csefind (struct fcrypt ∗, u_int) [static]

Definition at line 654 of file cryptodev.c.

Referenced by cryptof_ioctl().

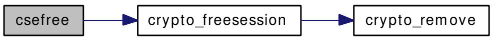
7.8.1.21 static int csefree (struct csession ∗) [static]

Definition at line 718 of file cryptodev.c.

References crypto_freesession().

Referenced by cryptof_close(), and cryptof_ioctl().

Here is the call graph for this function:



7.8.1.22 DECLARE_MODULE (cryptodev, cryptodev_mod, SI_SUB_PSEUDO, SI_ORDER_ANY)

7.8.1.23 MODULE_DEPEND (cryptodev, zlib, 1, 1, 1)

7.8.1.24 MODULE_DEPEND (cryptodev, crypto, 1, 1, 1)

7.8.1.25 MODULE_VERSION (cryptodev, 1)

7.8.2 Variable Documentation

7.8.2.1 struct cdevsw crypto_cdevsw [static]

Initial value:

```
 {
        .d_version =    D_VERSION,
        .d_flags =      D_NEEDGIANT,
        .d_open =       cryptoopen,
        .d_read =       cryptoread,
        .d_write =      cryptowrite,
        .d_ioctl =      cryptoioctl,
        .d_name =       "crypto",
}
```

Definition at line 785 of file cryptodev.c.

Referenced by cryptodev_modevent().

7.8.2.2 struct cdev∗ crypto_dev [static]

Definition at line 794 of file cryptodev.c.

Referenced by cryptodev_modevent().

7.8.2.3 moduledata_t cryptodev_mod [static]

Initial value:

```
 {
        "cryptodev",
        cryptodev_modevent,
        0
}
```

Definition at line 818 of file cryptodev.c.

7.8.2.4 struct fileops cryptofops [static]

Initial value:

```
 {
    .fo_read = cryptof_rw,
    .fo_write = cryptof_rw,
    .fo_ioctl = cryptof_ioctl,
    .fo_poll = cryptof_poll,
    .fo_kqfilter = cryptof_kqfilter,
    .fo_stat = cryptof_stat,
    .fo_close = cryptof_close
}
```

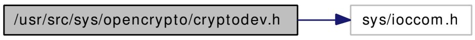
Definition at line 95 of file cryptodev.c.

Referenced by cryptoioctl().

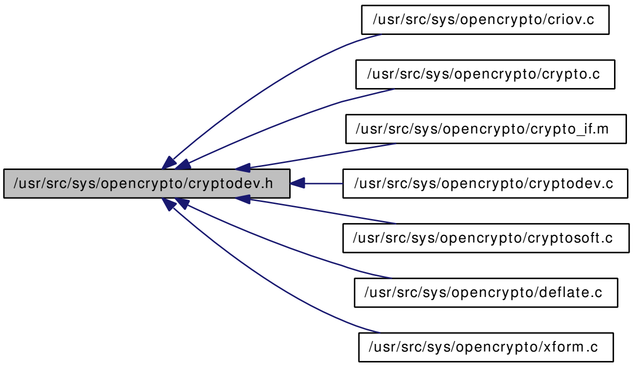
7.9 /usr/src/sys/opencrypto/cryptodev.h File Reference

```
#include <sys/ioccom.h>
```

Include dependency graph for cryptodev.h:



This graph shows which files directly or indirectly include this file:


/usr/src/sys/opencrypto/criov.c
/usr/src/sys/opencrypto/crypto.c
/usr/src/sys/opencrypto/crypto_if.m
/usr/src/sys/opencrypto/cryptodev.h
/usr/src/sys/opencrypto/cryptodev.c
/usr/src/sys/opencrypto/cryptosoft.c
/usr/src/sys/opencrypto/deflate.c
/usr/src/sys/opencrypto/xform.c

Data Structures

- struct session_op
- struct crypt_op
- struct crparam
- struct crypt_kop
- struct cryptotstat
- struct cryptostats
- struct cryptoini
- struct cryptodesc
- struct cryptop
- struct cryptkop
- struct cryptocap

Defines

- #define CRYPTO_DRIVERS_INITIAL 4
- #define CRYPTO_SW_SESSIONS 32
- #define NULL_HASH_LEN 16
- #define MD5_HASH_LEN 16
- #define SHA1_HASH_LEN 20
- #define RIPEMD160_HASH_LEN 20
- #define SHA2_256_HASH_LEN 32
- #define SHA2_384_HASH_LEN 48
- #define SHA2_512_HASH_LEN 64
- #define MD5_KPDK_HASH_LEN 16

- #define SHA1_KPDK_HASH_LEN 20
- #define HASH_MAX_LEN SHA2_512_HASH_LEN
- #define NULL_HMAC_BLOCK_LEN 64
- #define MD5_HMAC_BLOCK_LEN 64
- #define SHA1_HMAC_BLOCK_LEN 64
- #define RIPEMD160_HMAC_BLOCK_LEN 64
- #define SHA2_256_HMAC_BLOCK_LEN 64
- #define SHA2_384_HMAC_BLOCK_LEN 128
- #define SHA2_512_HMAC_BLOCK_LEN 128
- #define HMAC_MAX_BLOCK_LEN SHA2_512_HMAC_BLOCK_LEN
- #define HMAC_IPAD_VAL 0x36
- #define HMAC_OPAD_VAL 0x5C
- #define NULL_BLOCK_LEN 4
- #define DES_BLOCK_LEN 8
- #define DES3_BLOCK_LEN 8
- #define BLOWFISH_BLOCK_LEN 8
- #define SKIPJACK_BLOCK_LEN 8
- #define CAST128_BLOCK_LEN 8
- #define RIJNDAEL128_BLOCK_LEN 16
- #define AES_BLOCK_LEN RIJNDAEL128_BLOCK_LEN
- #define EALG_MAX_BLOCK_LEN AES_BLOCK_LEN
- #define CRYPTO_ALGORITHM_MIN 1
- #define CRYPTO_DES_CBC 1
- #define CRYPTO_3DES_CBC 2
- #define CRYPTO_BLF_CBC 3
- #define CRYPTO_CAST_CBC 4
- #define CRYPTO_SKIPJACK_CBC 5
- #define CRYPTO_MD5_HMAC 6
- #define CRYPTO_SHA1_HMAC 7
- #define CRYPTO_RIPEMD160_HMAC 8
- #define CRYPTO_MD5_KPDK 9
- #define CRYPTO_SHA1_KPDK 10
- #define CRYPTO_RIJNDAEL128_CBC 11
- #define CRYPTO_AES_CBC 11
- #define CRYPTO_ARC4 12
- #define CRYPTO_MD5 13
- #define CRYPTO_SHA1 14
- #define CRYPTO_NULL_HMAC 15
- #define CRYPTO_NULL_CBC 16
- #define CRYPTO_DEFLATE_COMP 17
- #define CRYPTO_SHA2_256_HMAC 18
- #define CRYPTO_SHA2_384_HMAC 19
- #define CRYPTO_SHA2_512_HMAC 20
- #define CRYPTO_ALGORITHM_MAX 20
- #define CRYPTO_ALG_FLAG_SUPPORTED 0x01
- #define CRYPTO_ALG_FLAG_RNG_ENABLE 0x02
- #define CRYPTO_ALG_FLAG_DSA_SHA 0x04
- #define COP_ENCRYPT 1
- #define COP_DECRYPT 2
- #define COP_F_BATCH 0x0008

- #define CRK_MAXPARAM 8
- #define CRK_ALGORITM_MIN 0
- #define CRK_MOD_EXP 0
- #define CRK_MOD_EXP_CRT 1
- #define CRK_DSA_SIGN 2
- #define CRK_DSA_VERIFY 3
- #define CRK_DH_COMPUTE_KEY 4
- #define CRK_ALGORITHM_MAX 4
- #define CRF_MOD_EXP (1 << CRK_MOD_EXP)
- #define CRF_MOD_EXP_CRT (1 << CRK_MOD_EXP_CRT)
- #define CRF_DSA_SIGN (1 << CRK_DSA_SIGN)
- #define CRF_DSA_VERIFY (1 << CRK_DSA_VERIFY)
- #define CRF_DH_COMPUTE_KEY (1 << CRK_DH_COMPUTE_KEY)
- #define CRIOGET _IOWR('c', 100, u_int32_t)
- #define CIOCGSESSION _IOWR('c', 101, struct session_op)
- #define CIOCFSESSION _IOW('c', 102, u_int32_t)
- #define CIOCCRYPT _IOWR('c', 103, struct crypt_op)
- #define CIOCKEY _IOWR('c', 104, struct crypt_kop)
- #define CIOCASYMFEAT _IOR('c', 105, u_int32_t)
- #define CRD_F_ENCRYPT 0x01
- #define CRD_F_IV_PRESENT 0x02
- #define CRD_F_IV_EXPLICIT 0x04
- #define CRD_F_DSA_SHA_NEEDED 0x08
- #define CRD_F_KEY_EXPLICIT 0x10
- #define CRD_F_COMP 0x0f
- #define crd_iv CRD_INI.cri_iv
- #define crd_key CRD_INI.cri_key
- #define crd_alg CRD_INI.cri_alg
- #define crd_klen CRD_INI.cri_klen
- #define CRYPTO_F_IMBUF 0x0001
- #define CRYPTO_F_IOV 0x0002
- #define CRYPTO_F_REL 0x0004
- #define CRYPTO_F_BATCH 0x0008
- #define CRYPTO_F_CBIMM 0x0010
- #define CRYPTO_F_DONE 0x0020
- #define CRYPTO_F_CBIFSYNC 0x0040
- #define CRYPTO_BUF_CONTIG 0x0
- #define CRYPTO_BUF_IOV 0x1
- #define CRYPTO_BUF_MBUF 0x2
- #define CRYPTO_OP_DECRYPT 0x0
- #define CRYPTO_OP_ENCRYPT 0x1
- #define CRYPTO_HINT_MORE 0x1
- #define CRYPTOCAP_F_CLEANUP 0x01
- #define CRYPTOCAP_F_SOFTWARE 0x02
- #define CRYPTOCAP_F_SYNC 0x04
- #define CRYPTO_SESID2HID(_sid) (((_sid) >> 32) & 0xffffff)
- #define CRYPTO_SESID2CAPS(_sid) (((_sid) >> 56) & 0xff)
- #define CRYPTO_SESID2LID(_sid) (((u_int32_t) (_sid)) & 0xffffffff)
- #define CRYPTO_SYMQ 0x1
- #define CRYPTO_ASYMQ 0x2

Functions

- MALLOC_DECLARE (M_CRYPTO_DATA)
- int crypto_newsession (u_int64_t ∗sid, struct cryptoini ∗cri, int hard)
- int crypto_freesession (u_int64_t sid)
- int32_t crypto_get_driverid (u_int32_t flags)
- int crypto_register (u_int32_t driverid, int alg, u_int16_t maxoplen, u_int32_t flags, int(∗newses)(void ∗, u_int32_t ∗, struct cryptoini ∗), int(∗freeses)(void ∗, u_int64_t), int(∗process)(void ∗, struct cryptop ∗, int), void ∗arg)
- int crypto_kregister (u_int32_t, int, u_int32_t, int(∗)(void ∗, struct cryptkop ∗, int), void ∗arg)
- int crypto_unregister (u_int32_t driverid, int alg)
- int crypto_unregister_all (u_int32_t driverid)
- int crypto_dispatch (struct cryptop ∗crp)
- int crypto_kdispatch (struct cryptkop ∗)
- int crypto_unblock (u_int32_t, int)
- void crypto_done (struct cryptop ∗crp)
- void crypto_kdone (struct cryptkop ∗)
- int crypto_getfeat (int ∗)
- void crypto_freereq (struct cryptop ∗crp)
- cryptop ∗ crypto_getreq (int num)
- void cuio_copydata (struct uio ∗uio, int off, int len, caddr_t cp)
- void cuio_copyback (struct uio ∗uio, int off, int len, caddr_t cp)
- iovec ∗ cuio_getptr (struct uio ∗uio, int loc, int ∗off)
- int cuio_apply (struct uio ∗uio, int off, int len, int(∗f)(void ∗, void ∗, u_int), void ∗arg)
- void crypto_copyback (int flags, caddr_t buf, int off, int size, caddr_t in)
- void crypto_copydata (int flags, caddr_t buf, int off, int size, caddr_t out)
- int crypto_apply (int flags, caddr_t buf, int off, int len, int(∗f)(void ∗, void ∗, u_int), void ∗arg)

Variables

- int crypto_usercrypto
- int crypto_userasymcrypto
- int crypto_devallowsoft

7.9.1 Define Documentation

7.9.1.1 #define AES_BLOCK_LEN RIJNDAEL128_BLOCK_LEN

Definition at line 98 of file cryptodev.h.

7.9.1.2 #define BLOWFISH_BLOCK_LEN 8

Definition at line 94 of file cryptodev.h.

7.9.1.3 #define CAST128_BLOCK_LEN 8

Definition at line 96 of file cryptodev.h.

7.9.1.4 #define CIOCASYMFEAT _IOR('c', 105, u_int32_t)

Definition at line 197 of file cryptodev.h.

Referenced by cryptof_ioctl().

7.9.1.5 #define CIOCCRYPT _IOWR('c', 103, struct crypt_op)

Definition at line 194 of file cryptodev.h.

Referenced by cryptof_ioctl().

7.9.1.6 #define CIOCFSESSION _IOW('c', 102, u_int32_t)

Definition at line 193 of file cryptodev.h.

Referenced by cryptof_ioctl().

7.9.1.7 #define CIOCGSESSION _IOWR('c', 101, struct session_op)

Definition at line 192 of file cryptodev.h.

Referenced by cryptof_ioctl().

7.9.1.8 #define CIOCKEY _IOWR('c', 104, struct crypt_kop)

Definition at line 195 of file cryptodev.h.

Referenced by cryptof_ioctl().

7.9.1.9 #define COP_DECRYPT 2

Definition at line 146 of file cryptodev.h.

7.9.1.10 #define COP_ENCRYPT 1

Definition at line 145 of file cryptodev.h.

Referenced by cryptodev_op().

7.9.1.11 #define COP_F_BATCH 0x0008

Definition at line 148 of file cryptodev.h.

Referenced by cryptodev_op().

7.9.1.12 #define crd_alg CRD_INI.cri_alg

Definition at line 257 of file cryptodev.h.

7.9.1.13 #define CRD_F_COMP 0x0f

Definition at line 252 of file cryptodev.h.

Referenced by swcr_compdec().

7.9.1.14 #define CRD_F_DSA_SHA_NEEDED 0x08

Definition at line 250 of file cryptodev.h.

7.9.1.15 #define CRD_F_ENCRYPT 0x01

Definition at line 246 of file cryptodev.h.

Referenced by cryptodev_op(), and swcr_encdec().

7.9.1.16 #define CRD_F_IV_EXPLICIT 0x04

Definition at line 249 of file cryptodev.h.

Referenced by cryptodev_op(), and swcr_encdec().

7.9.1.17 #define CRD_F_IV_PRESENT 0x02

Definition at line 247 of file cryptodev.h.

Referenced by cryptodev_op(), and swcr_encdec().

7.9.1.18 #define CRD_F_KEY_EXPLICIT 0x10

Definition at line 251 of file cryptodev.h.

Referenced by swcr_authcompute(), and swcr_encdec().

7.9.1.19 #define crd_iv CRD_INI.cri_iv

Definition at line 255 of file cryptodev.h.

7.9.1.20 #define crd_key CRD_INI.cri_key

Definition at line 256 of file cryptodev.h.

7.9.1.21 #define crd_klen CRD_INI.cri_klen

Definition at line 258 of file cryptodev.h.

7.9.1.22 #define CRF_DH_COMPUTE_KEY (1 << CRK_DH_COMPUTE_KEY)

Definition at line 183 of file cryptodev.h.

7.9.1.23 #define CRF_DSA_SIGN (1 << CRK_DSA_SIGN)

Definition at line 181 of file cryptodev.h.

7.9.1.24 #define CRF_DSA_VERIFY (1 << CRK_DSA_VERIFY)

Definition at line 182 of file cryptodev.h.

7.9.1.25 #define CRF_MOD_EXP (1 << CRK_MOD_EXP)

Definition at line 179 of file cryptodev.h.

7.9.1.26 #define CRF_MOD_EXP_CRT (1 << CRK_MOD_EXP_CRT)

Definition at line 180 of file cryptodev.h.

7.9.1.27 #define CRIOGET _IOWR('c', 100, u_int32_t)

Definition at line 189 of file cryptodev.h.

Referenced by cryptoioctl().

7.9.1.28 #define CRK_ALGORITHM_MAX 4

Definition at line 177 of file cryptodev.h.

Referenced by crypto_getfeat(), and crypto_kregister().

7.9.1.29 #define CRK_ALGORITM_MIN 0

Definition at line 171 of file cryptodev.h.

Referenced by crypto_kregister().

7.9.1.30 #define CRK_DH_COMPUTE_KEY 4

Definition at line 176 of file cryptodev.h.

Referenced by cryptodev_key().

7.9.1.31 #define CRK_DSA_SIGN 2

Definition at line 174 of file cryptodev.h.

Referenced by cryptodev_key().

7.9.1.32 #define CRK_DSA_VERIFY 3

Definition at line 175 of file cryptodev.h.

Referenced by cryptodev_key().

7.9.1.33 #define CRK_MAXPARAM 8

Definition at line 161 of file cryptodev.h.

Referenced by cryptodev_key().

7.9.1.34 #define CRK_MOD_EXP 0

Definition at line 172 of file cryptodev.h.

Referenced by cryptodev_key().

7.9.1.35 #define CRK_MOD_EXP_CRT 1

Definition at line 173 of file cryptodev.h.

Referenced by cryptodev_key().

7.9.1.36 #define CRYPTO_3DES_CBC 2

Definition at line 103 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.37 #define CRYPTO_AES_CBC 11

Definition at line 113 of file cryptodev.h.

Referenced by cryptof_ioctl().

7.9.1.38 #define CRYPTO_ALG_FLAG_DSA_SHA 0x04

Definition at line 128 of file cryptodev.h.

7.9.1.39 #define CRYPTO_ALG_FLAG_RNG_ENABLE 0x02

Definition at line 127 of file cryptodev.h.

7.9.1.40 #define CRYPTO_ALG_FLAG_SUPPORTED 0x01

Definition at line 126 of file cryptodev.h.

Referenced by crypto_getfeat(), crypto_kinvoke(), crypto_kregister(), and crypto_register().

7.9.1.41 #define CRYPTO_ALGORITHM_MAX 20

Definition at line 123 of file cryptodev.h.

Referenced by crypto_register(), crypto_unregister(), and crypto_unregister_all().

7.9.1.42 #define CRYPTO_ALGORITHM_MIN 1

Definition at line 101 of file cryptodev.h.

Referenced by crypto_register(), crypto_unregister(), and crypto_unregister_all().

7.9.1.43 #define CRYPTO_ARC4 12

Definition at line 114 of file cryptodev.h.

Referenced by cryptodev_op(), and cryptof_ioctl().

7.9.1.44 #define CRYPTO_ASYMQ 0x2

Definition at line 391 of file cryptodev.h.

Referenced by crypto_unblock().

7.9.1.45 #define CRYPTO_BLF_CBC 3

Definition at line 104 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.46 #define CRYPTO_BUF_CONTIG 0x0

Definition at line 300 of file cryptodev.h.

7.9.1.47 #define CRYPTO_BUF_IOV 0x1

Definition at line 301 of file cryptodev.h.

7.9.1.48 #define CRYPTO_BUF_MBUF 0x2

Definition at line 302 of file cryptodev.h.

7.9.1.49 #define CRYPTO_CAST_CBC 4

Definition at line 105 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.50 #define CRYPTO_DEFLATE_COMP 17

Definition at line 119 of file cryptodev.h.

Referenced by swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.51 #define CRYPTO_DES_CBC 1

Definition at line 102 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.52 #define CRYPTO_DRIVERS_INITIAL 4

Definition at line 61 of file cryptodev.h.

7.9.1.53 #define CRYPTO_F_BATCH 0x0008

Definition at line 286 of file cryptodev.h.

Referenced by crypto_dispatch(), and crypto_proc().

7.9.1.54 #define CRYPTO_F_CBIFSYNC 0x0040

Definition at line 289 of file cryptodev.h.

Referenced by crypto_done().

7.9.1.55 #define CRYPTO_F_CBIMM 0x0010

Definition at line 287 of file cryptodev.h.

Referenced by crypto_done(), and cryptodev_op().

7.9.1.56 #define CRYPTO_F_DONE 0x0020

Definition at line 288 of file cryptodev.h.

Referenced by crypto_done(), and cryptodev_op().

7.9.1.57 #define CRYPTO_F_IMBUF 0x0001

Definition at line 283 of file cryptodev.h.

Referenced by crypto_apply(), crypto_copyback(), crypto_copydata(), swcr_compdec(), and swcr_encdec().

7.9.1.58 #define CRYPTO_F_IOV 0x0002

Definition at line 284 of file cryptodev.h.

Referenced by crypto_apply(), crypto_copyback(), crypto_copydata(), cryptodev_op(), swcr_compdec(), and swcr_encdec().

7.9.1.59 #define CRYPTO_F_REL 0x0004

Definition at line 285 of file cryptodev.h.

7.9.1.60 #define CRYPTO_HINT_MORE 0x1

Definition at line 310 of file cryptodev.h.

Referenced by crypto_proc().

7.9.1.61 #define CRYPTO_MD5 13

Definition at line 115 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.62 #define CRYPTO_MD5_HMAC 6

Definition at line 107 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.63 #define CRYPTO_MD5_KPDK 9

Definition at line 110 of file cryptodev.h.

Referenced by swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.64 #define CRYPTO_NULL_CBC 16

Definition at line 118 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.65 #define CRYPTO_NULL_HMAC 15

Definition at line 117 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.66 #define CRYPTO_OP_DECRYPT 0x0

Definition at line 304 of file cryptodev.h.

7.9.1.67 #define CRYPTO_OP_ENCRYPT 0x1

Definition at line 305 of file cryptodev.h.

7.9.1.68 #define CRYPTO_RIJNDAEL128_CBC 11

Definition at line 112 of file cryptodev.h.

Referenced by swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.69 #define CRYPTO_RIPEMD160_HMAC 8

Definition at line 109 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.70 #define CRYPTO_SESID2CAPS(_sid) (((_sid) >> 56) & 0xff)

Definition at line 369 of file cryptodev.h.

Referenced by crypto_done().

7.9.1.71 #define CRYPTO_SESID2HID(_sid) (((_sid) >> 32) & 0xffffff)

Definition at line 368 of file cryptodev.h.

Referenced by crypto_dispatch(), crypto_freesession(), and crypto_proc().

7.9.1.72 #define CRYPTO_SESID2LID(_sid) (((u_int32_t) (_sid)) & 0xffffffff)

Definition at line 370 of file cryptodev.h.

Referenced by swcr_freesession().

7.9.1.73 #define CRYPTO_SHA1 14

Definition at line 116 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.74 #define CRYPTO_SHA1_HMAC 7

Definition at line 108 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.75 #define CRYPTO_SHA1_KPDK 10

Definition at line 111 of file cryptodev.h.

Referenced by swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_-newsession(), and swcr_process().

7.9.1.76 #define CRYPTO_SHA2_256_HMAC 18

Definition at line 120 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.77 #define CRYPTO_SHA2_384_HMAC 19

Definition at line 121 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.78 #define CRYPTO_SHA2_512_HMAC 20

Definition at line 122 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_authcompute(), swcr_authprepare(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.79 #define CRYPTO_SKIPJACK_CBC 5

Definition at line 106 of file cryptodev.h.

Referenced by cryptof_ioctl(), swcr_freesession(), swcr_init(), swcr_newsession(), and swcr_process().

7.9.1.80 #define CRYPTO_SW_SESSIONS 32

Definition at line 62 of file cryptodev.h.

Referenced by swcr_newsession().

7.9.1.81 #define CRYPTO_SYMQ 0x1

Definition at line 390 of file cryptodev.h.

Referenced by crypto_unblock().

7.9.1.82 #define CRYPTOCAP_F_CLEANUP 0x01

Definition at line 347 of file cryptodev.h.

Referenced by crypto_freesession(), crypto_get_driverid(), crypto_invoke(), crypto_kdone(), crypto_newsession(), crypto_unregister(), and crypto_unregister_all().

7.9.1.83 #define CRYPTOCAP_F_SOFTWARE 0x02

Definition at line 348 of file cryptodev.h.

Referenced by crypto_getfeat(), crypto_kinvoke(), crypto_newsession(), and swcr_init().

7.9.1.84 #define CRYPTOCAP_F_SYNC 0x04

Definition at line 349 of file cryptodev.h.

Referenced by crypto_done(), and swcr_init().

7.9.1.85 #define DES3_BLOCK_LEN 8

Definition at line 93 of file cryptodev.h.

7.9.1.86 #define DES_BLOCK_LEN 8

Definition at line 92 of file cryptodev.h.

7.9.1.87 #define EALG_MAX_BLOCK_LEN AES_BLOCK_LEN

Definition at line 99 of file cryptodev.h.

Referenced by swcr_encdec().

7.9.1.88 #define HASH_MAX_LEN SHA2_512_HASH_LEN

Definition at line 75 of file cryptodev.h.

Referenced by swcr_authcompute().

7.9.1.89 #define HMAC_IPAD_VAL 0x36

Definition at line 87 of file cryptodev.h.

Referenced by swcr_authprepare(), and swcr_init().

7.9.1.90 #define HMAC_MAX_BLOCK_LEN SHA2_512_HMAC_BLOCK_LEN

Definition at line 86 of file cryptodev.h.

Referenced by swcr_init().

7.9.1.91 #define HMAC_OPAD_VAL 0x5C

Definition at line 88 of file cryptodev.h.

Referenced by swcr_authprepare(), and swcr_init().

7.9.1.92 #define MD5_HASH_LEN 16

Definition at line 66 of file cryptodev.h.

7.9.1.93 #define MD5_HMAC_BLOCK_LEN 64

Definition at line 79 of file cryptodev.h.

7.9.1.94 #define MD5_KPDK_HASH_LEN 16

Definition at line 72 of file cryptodev.h.

7.9.1.95 #define NULL_BLOCK_LEN 4

Definition at line 91 of file cryptodev.h.

7.9.1.96 #define NULL_HASH_LEN 16

Definition at line 65 of file cryptodev.h.

7.9.1.97 #define NULL_HMAC_BLOCK_LEN 64

Definition at line 78 of file cryptodev.h.

7.9.1.98 #define RIJNDAEL128_BLOCK_LEN 16

Definition at line 97 of file cryptodev.h.

7.9.1.99 #define RIPEMD160_HASH_LEN 20

Definition at line 68 of file cryptodev.h.

7.9.1.100 #define RIPEMD160_HMAC_BLOCK_LEN 64

Definition at line 81 of file cryptodev.h.

7.9.1.101 #define SHA1_HASH_LEN 20

Definition at line 67 of file cryptodev.h.

7.9.1.102 #define SHA1_HMAC_BLOCK_LEN 64

Definition at line 80 of file cryptodev.h.

7.9.1.103 #define SHA1_KPDK_HASH_LEN 20

Definition at line 73 of file cryptodev.h.

7.9.1.104 #define SHA2_256_HASH_LEN 32

Definition at line 69 of file cryptodev.h.

7.9.1.105 #define SHA2_256_HMAC_BLOCK_LEN 64

Definition at line 82 of file cryptodev.h.

7.9.1.106 #define SHA2_384_HASH_LEN 48

Definition at line 70 of file cryptodev.h.

7.9.1.107 #define SHA2_384_HMAC_BLOCK_LEN 128

Definition at line 83 of file cryptodev.h.

7.9.1.108 #define SHA2_512_HASH_LEN 64

Definition at line 71 of file cryptodev.h.

7.9.1.109 #define SHA2_512_HMAC_BLOCK_LEN 128

Definition at line 84 of file cryptodev.h.

7.9.1.110 #define SKIPJACK_BLOCK_LEN 8

Definition at line 95 of file cryptodev.h.

7.9.2 Function Documentation

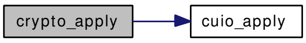
7.9.2.1 int crypto_apply (int *flags*, caddr_t *buf*, int *off*, int *len*, int(∗)(void ∗, void ∗, u_int) *f*, void ∗ *arg*)

Definition at line 186 of file criov.c.

References CRYPTO_F_IMBUF, CRYPTO_F_IOV, and cuio_apply().

Referenced by swcr_authcompute().

Here is the call graph for this function:



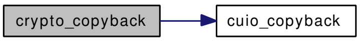
7.9.2.2 void crypto_copyback (int *flags*, caddr_t *buf*, int *off*, int *size*, caddr_t *in*)

Definition at line 162 of file criov.c.

References CRYPTO_F_IMBUF, CRYPTO_F_IOV, and cuio_copyback().

Referenced by swcr_authcompute(), swcr_compdec(), and swcr_encdec().

Here is the call graph for this function:



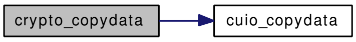
7.9.2.3 void crypto_copydata (int *flags*, caddr_t *buf*, int *off*, int *size*, caddr_t *out*)

Definition at line 174 of file criov.c.

References CRYPTO_F_IMBUF, CRYPTO_F_IOV, and cuio_copydata().

Referenced by swcr_compdec(), and swcr_encdec().

Here is the call graph for this function:



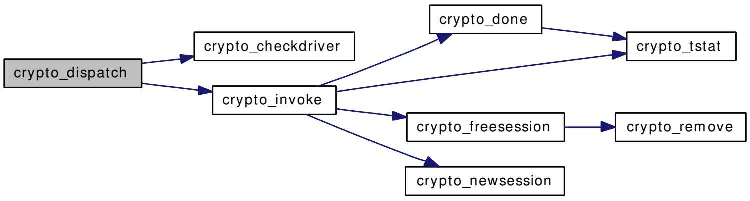
7.9.2.4 int crypto_dispatch (struct cryptop ∗ *crp*)

Definition at line 701 of file crypto.c.

References cryptocap::cc_qblocked, cryptop::crp_flags, cryptop::crp_sid, crp_sleep, cryptop::crp_tstamp, crypto_checkdriver(), CRYPTO_F_BATCH, crypto_invoke(), CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, CRYPTO_SESID2HID, and cryptostats::cs_ops.

Referenced by cryptodev_cb(), and cryptodev_op().

Here is the call graph for this function:



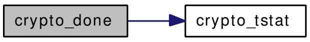
7.9.2.5 void crypto_done (struct cryptop ∗ *crp*)

Definition at line 964 of file crypto.c.

References cryptop::crp_callback, cryptop::crp_etype, cryptop::crp_flags, cryptop::crp_sid, cryptop::crp_tstamp, CRYPTO_F_CBIFSYNC, CRYPTO_F_CBIMM, CRYPTO_F_DONE, CRYPTO_RETQ_EMPTY, CRYPTO_RETQ_LOCK, CRYPTO_RETQ_UNLOCK, CRYPTO_SESID2CAPS, crypto_tstat(), CRYPTOCAP_F_SYNC, cryptostats::cs_cb, cryptostats::cs_done, cryptostats::cs_errs, and cryptostats::cs_finis.

Referenced by crypto_invoke().

Here is the call graph for this function:



7.9.2.6 void crypto_freereq (struct cryptop ∗ *crp*)

Definition at line 899 of file crypto.c.

References cryptodesc::crd_next, cryptop::crp_desc, CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, CRYPTO_RETQ_LOCK, and CRYPTO_RETQ_UNLOCK.

Referenced by crypto_getreq(), and cryptodev_op().

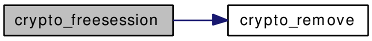
7.9.2.7 int crypto_freesession (u_int64_t *sid*)

Definition at line 383 of file crypto.c.

References cryptocap::cc_arg, cryptocap::cc_flags, cryptocap::cc_freesession, cryptocap::cc_sessions, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, crypto_remove(), CRYPTO_SESID2HID, and CRYPTOCAP_F_CLEANUP.

Referenced by crypto_invoke(), cryptof_ioctl(), and csefree().

Here is the call graph for this function:



7.9.2.8 int32_t crypto_get_driverid (u_int32_t *flags*)

Definition at line 427 of file crypto.c.

References cryptocap::cc_flags, cryptocap::cc_process, cryptocap::cc_sessions, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, and CRYPTOCAP_F_CLEANUP.

Referenced by swcr_init().

7.9.2.9 int crypto_getfeat (int ∗)

Definition at line 1044 of file crypto.c.

References cryptocap::cc_flags, cryptocap::cc_kalg, cryptocap::cc_kprocess, CRK_ALGORITHM_MAX, CRYPTO_ALG_FLAG_SUPPORTED, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, and CRYPTOCAP_F_SOFTWARE.

Referenced by cryptof_ioctl().

7.9.2.10 struct cryptop∗ crypto_getreq (int *num*)

Definition at line 939 of file crypto.c.

References cryptodesc::crd_next, cryptop::crp_desc, and crypto_freereq().

Referenced by cryptodev_op().

Here is the call graph for this function:

crypto_getreq
crypto_freereq

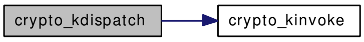
7.9.2.11 int crypto_kdispatch (struct cryptkop *)

Definition at line 748 of file crypto.c.

References crp_sleep, crypto_kinvoke(), CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, and cryptostats::cs_kops.

Referenced by cryptodev_key().

Here is the call graph for this function:

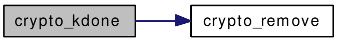


7.9.2.12 void crypto_kdone (struct cryptkop *)

Definition at line 1020 of file crypto.c.

References cryptocap::cc_flags, cryptocap::cc_koperations, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, crypto_remove(), CRYPTO_RETQ_EMPTY, CRYPTO_RETQ_LOCK, CRYPTO_RETQ_UNLOCK, CRYPTOCAP_F_CLEANUP, cryptostats::cs_kerrs, cryptkop::krp_hid, and cryptkop::krp_status.

Here is the call graph for this function:

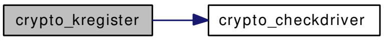


7.9.2.13 int crypto_kregister (u_int32_t, int, u_int32_t, int(*)(void *, struct cryptkop *, int), void * *arg*)

Definition at line 491 of file crypto.c.

References cryptocap::cc_kalg, cryptocap::cc_karg, cryptocap::cc_kprocess, CRK_ALGORITHM_MAX, CRK_ALGORITM_MIN, CRYPTO_ALG_FLAG_SUPPORTED, crypto_checkdriver(), CRYPTO_DRIVER_LOCK, and CRYPTO_DRIVER_UNLOCK.

Here is the call graph for this function:



7.9.2.14 int crypto_newsession (u_int64_t * *sid*, struct cryptoini * *cri*, int *hard*)

Definition at line 263 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_arg, cryptocap::cc_flags, cryptocap::cc_newsession, cryptocap::cc_sessions, cryptoini::cri_alg, cryptoini::cri_next, CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, crypto_drivers, crypto_drivers_num, CRYPTOCAP_F_CLEANUP, and CRYPTOCAP_F_SOFTWARE.

Referenced by crypto_invoke(), and cryptof_ioctl().

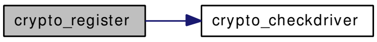
7.9.2.15 int crypto_register (u_int32_t *driverid*, int *alg*, u_int16_t *maxoplen*, u_int32_t *flags*, int(∗)(void ∗, u_int32_t ∗, struct cryptoini ∗) *newses*, int(∗)(void ∗, u_int64_t) *freeses*, int(∗)(void ∗, struct cryptop ∗, int) *process*, void ∗ *arg*)

Definition at line 534 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_arg, cryptocap::cc_freesession, cryptocap::cc_max_op_len, cryptocap::cc_newsession, cryptocap::cc_process, cryptocap::cc_sessions, CRYPTO_ALG_FLAG_SUPPORTED, CRYPTO_ALGORITHM_MAX, CRYPTO_ALGORITHM_MIN, crypto_checkdriver(), CRYPTO_DRIVER_LOCK, and CRYPTO_DRIVER_UNLOCK.

Referenced by swcr_init().

Here is the call graph for this function:

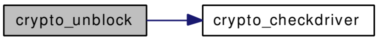


7.9.2.16 int crypto_unblock (u_int32_t, int)

Definition at line 675 of file crypto.c.

References cryptocap::cc_kqblocked, cryptocap::cc_qblocked, crp_sleep, CRYPTO_ASYMQ, crypto_checkdriver(), CRYPTO_Q_LOCK, CRYPTO_Q_UNLOCK, and CRYPTO_SYMQ.

Here is the call graph for this function:



7.9.2.17 int crypto_unregister (u_int32_t *driverid*, int *alg*)

Definition at line 588 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_koperations, cryptocap::cc_max_op_len, cryptocap::cc_sessions, CRYPTO_ALGORITHM_MAX, CRYPTO_ALGORITHM_MIN, crypto_checkdriver(), CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, and CRYPTOCAP_F_CLEANUP.

Here is the call graph for this function:

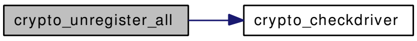
crypto_unregister
crypto_checkdriver

7.9.2.18 int crypto_unregister_all (u_int32_t *driverid*)

Definition at line 637 of file crypto.c.

References cryptocap::cc_alg, cryptocap::cc_koperations, cryptocap::cc_max_op_len, cryptocap::cc_sessions, CRYPTO_ALGORITHM_MAX, CRYPTO_ALGORITHM_MIN, crypto_checkdriver(), CRYPTO_DRIVER_LOCK, CRYPTO_DRIVER_UNLOCK, and CRYPTOCAP_F_CLEANUP.

Here is the call graph for this function:



7.9.2.19 int cuio_apply (struct uio ∗ *uio*, int *off*, int *len*, int(∗)(void ∗, void ∗, u_int) *f*, void ∗ *arg*)

Definition at line 138 of file criov.c.

References CUIO_SKIP.

Referenced by crypto_apply().

7.9.2.20 void cuio_copyback (struct uio ∗ *uio*, int *off*, int *len*, caddr_t *cp*)

Definition at line 82 of file criov.c.

References CUIO_SKIP.

Referenced by crypto_copyback(), and swcr_encdec().

7.9.2.21 void cuio_copydata (struct uio ∗ *uio*, int *off*, int *len*, caddr_t *cp*)

Definition at line 62 of file criov.c.

References CUIO_SKIP.

Referenced by crypto_copydata(), and swcr_encdec().

7.9.2.22 struct iovec∗ cuio_getptr (struct uio ∗ *uio*, int *loc*, int ∗ *off*)

Definition at line 105 of file criov.c.

Referenced by swcr_encdec().

7.9.2.23 MALLOC_DECLARE (M_CRYPTO_DATA)

7.9.3 Variable Documentation

7.9.3.1 int crypto_devallowsoft

Referenced by cryptof_ioctl(), and TAILQ_HEAD().

7.9.3.2 int crypto_userasymcrypto

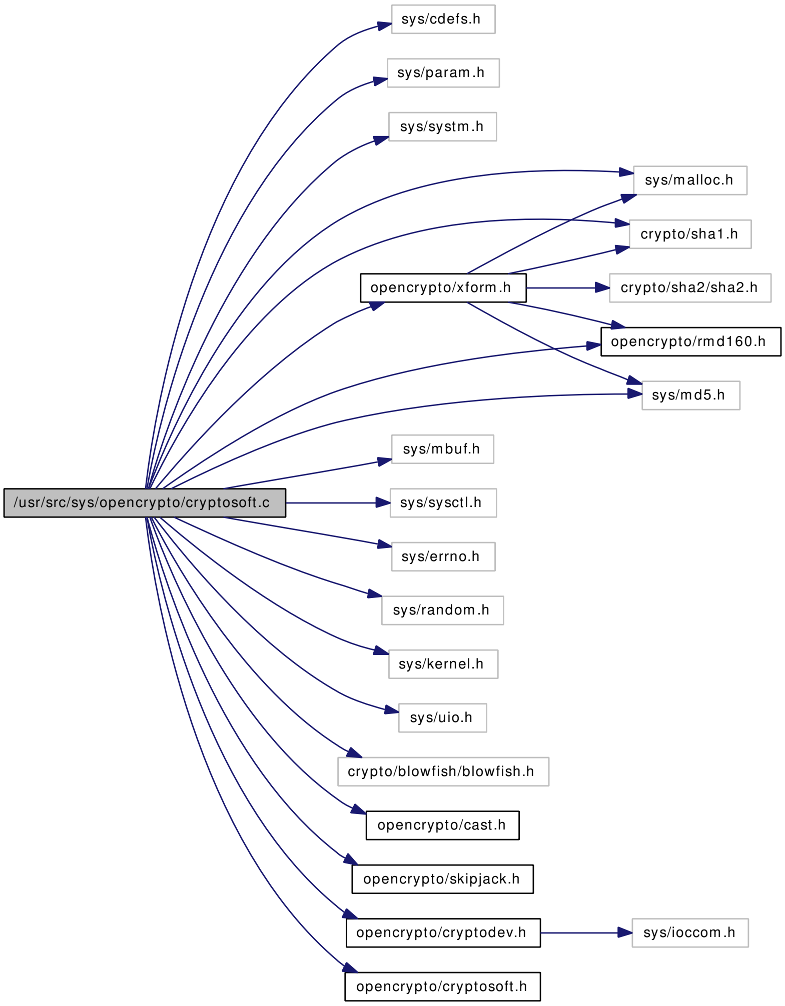
Referenced by TAILQ_HEAD().

7.9.3.3 int crypto_usercrypto

7.10 /usr/src/sys/opencrypto/cryptosoft.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/systm.h>
#include <sys/malloc.h>
#include <sys/mbuf.h>
#include <sys/sysctl.h>
#include <sys/errno.h>
#include <sys/random.h>
#include <sys/kernel.h>
#include <sys/uio.h>
#include <crypto/blowfish/blowfish.h>
#include <crypto/sha1.h>
#include <opencrypto/rmd160.h>
#include <opencrypto/cast.h>
#include <opencrypto/skipjack.h>
#include <sys/md5.h>
#include <opencrypto/cryptodev.h>
#include <opencrypto/cryptosoft.h>
#include <opencrypto/xform.h>
```

Include dependency graph for cryptosoft.c:



Defines

- #define REGISTER(alg) crypto_register(swcr_id, alg, 0,0,NULL,NULL,NULL,NULL)

Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/cryptosoft.c,v 1.17 2006/06/04 22:17:25 pjd Exp $")
- static int swcr_encdec (struct cryptodesc ∗, struct swcr_data ∗, caddr_t, int)
- static int swcr_authcompute (struct cryptodesc ∗, struct swcr_data ∗, caddr_t, int)
- static int swcr_compdec (struct cryptodesc ∗, struct swcr_data ∗, caddr_t, int)
- static int swcr_process (void ∗, struct cryptop ∗, int)
- static int swcr_newsession (void ∗, u_int32_t ∗, struct cryptoini ∗)
- static int swcr_freesession (void ∗, u_int64_t)
- static void swcr_authprepare (struct auth_hash ∗axf, struct swcr_data ∗sw, u_char ∗key, int klen)
- static void swcr_init (void)
- static void swcr_uninit (void)
- SYSUNINIT (cryptosoft_uninit, SI_SUB_PSEUDO, SI_ORDER_ANY, swcr_uninit, NULL)

Variables

- u_int8_t ∗ hmac_ipad_buffer
- u_int8_t ∗ hmac_opad_buffer
- swcr_data ∗∗ swcr_sessions = NULL
- u_int32_t swcr_sesnum = 0
- int32_t swcr_id = -1

7.10.1 Define Documentation

7.10.1.1 #define REGISTER(alg) crypto_register(swcr_id, alg, 0,0,NULL,NULL,NULL,NULL)

Referenced by swcr_init().

7.10.2 Function Documentation

7.10.2.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/cryptosoft. *c*, v 1.17 2006/06/04 22:17:25 pjd Exp $")

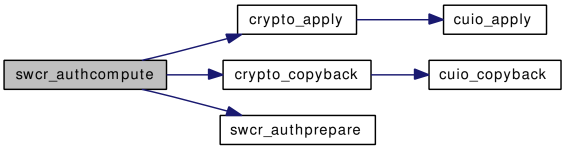
7.10.2.2 static int swcr_authcompute (struct cryptodesc ∗, struct swcr_data ∗, caddr_t, int) [static]

Definition at line 444 of file cryptosoft.c.

References CRD_F_KEY_EXPLICIT, cryptodesc::crd_flags, cryptodesc::crd_inject, cryptodesc::crd_len, cryptodesc::crd_skip, crypto_apply(), crypto_copyback(), CRYPTO_MD5_HMAC, CRYPTO_MD5_KPDK, CRYPTO_NULL_HMAC, CRYPTO_RIPEMD160_HMAC, CRYPTO_SHA1_HMAC, CRYPTO_SHA1_KPDK, CRYPTO_SHA2_256_HMAC, CRYPTO_SHA2_384_HMAC, CRYPTO_SHA2_512_HMAC, HASH_MAX_LEN, swcr_data::sw_alg, and swcr_authprepare().

Referenced by swcr_process().

Here is the call graph for this function:



7.10.2.3 static void swcr_authprepare (struct auth_hash ∗ *axf*, struct swcr_data ∗ *sw*, u_char ∗ *key*, int *klen*) [static]

Definition at line 394 of file cryptosoft.c.

References auth_hash::blocksize, CRYPTO_MD5_HMAC, CRYPTO_MD5_KPDK, CRYPTO_NULL_HMAC, CRYPTO_RIPEMD160_HMAC, CRYPTO_SHA1_HMAC, CRYPTO_SHA1_KPDK, CRYPTO_SHA2_256_HMAC, CRYPTO_SHA2_384_HMAC, CRYPTO_SHA2_512_HMAC, auth_hash::Final, hmac_ipad_buffer, HMAC_IPAD_VAL, hmac_opad_buffer, HMAC_OPAD_VAL, auth_hash::Init, auth_hash::type, and auth_hash::Update.

Referenced by swcr_authcompute(), and swcr_newsession().

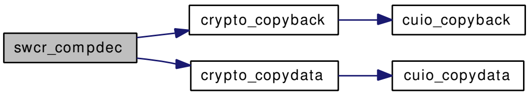
7.10.2.4 static int swcr_compdec (struct cryptodesc ∗, struct swcr_data ∗, caddr_t, int) [static]

Definition at line 507 of file cryptosoft.c.

References comp_algo::compress, CRD_F_COMP, cryptodesc::crd_flags, cryptodesc::crd_len, cryptodesc::crd_skip, crypto_copyback(), crypto_copydata(), CRYPTO_F_IMBUF, CRYPTO_F_IOV, and comp_algo::decompress.

Referenced by swcr_process().

Here is the call graph for this function:



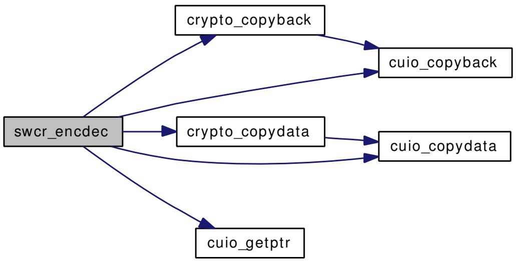
7.10.2.5 static int swcr_encdec (struct cryptodesc ∗, struct swcr_data ∗, caddr_t, int) [static]

Definition at line 66 of file cryptosoft.c.

References enc_xform::blocksize, CRD_F_ENCRYPT, CRD_F_IV_EXPLICIT, CRD_F_IV_-PRESENT, CRD_F_KEY_EXPLICIT, cryptodesc::crd_flags, cryptodesc::crd_inject, cryptodesc::crd_len, cryptodesc::crd_skip, crypto_copyback(), crypto_copydata(), CRYPTO_F_IMBUF, CRYPTO_F_IOV, cuio_copyback(), cuio_copydata(), cuio_getptr(), enc_xform::decrypt, EALG_MAX_BLOCK_LEN, enc_xform::encrypt, enc_xform::setkey, and enc_xform::zerokey.

Referenced by swcr_process().

Here is the call graph for this function:



7.10.2.6 static int swcr_freesession (void ∗, u_int64_t) [static]

Definition at line 789 of file cryptosoft.c.

References CRYPTO_3DES_CBC, CRYPTO_BLF_CBC, CRYPTO_CAST_CBC, CRYPTO_-DEFLATE_COMP, CRYPTO_DES_CBC, CRYPTO_MD5, CRYPTO_MD5_HMAC, CRYPTO_-MD5_KPDK, CRYPTO_NULL_CBC, CRYPTO_NULL_HMAC, CRYPTO_RIJNDAEL128_CBC, CRYPTO_RIPEMD160_HMAC, CRYPTO_SESID2LID, CRYPTO_SHA1, CRYPTO_SHA1_HMAC, CRYPTO_SHA1_KPDK, CRYPTO_SHA2_256_HMAC, CRYPTO_SHA2_384_HMAC, CRYPTO_-SHA2_512_HMAC, CRYPTO_SKIPJACK_CBC, auth_hash::ctxsize, swcr_data::sw_alg, swcr_-data::sw_next, swcr_sesnum, swcr_sessions, and enc_xform::zerokey.

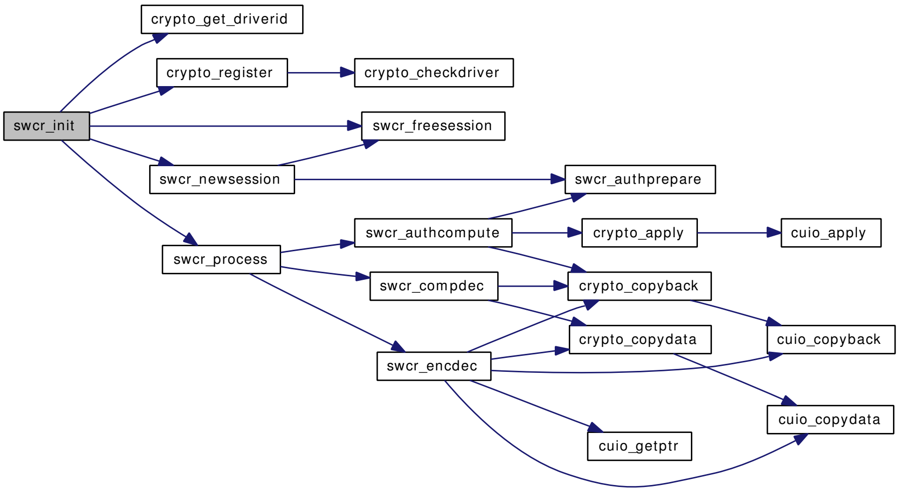
Referenced by swcr_init(), and swcr_newsession().

7.10.2.7 static void swcr_init (void) [static]

Definition at line 974 of file cryptosoft.c.

References CRYPTO_3DES_CBC, CRYPTO_BLF_CBC, CRYPTO_CAST_CBC, CRYPTO_DEFLATE_COMP, CRYPTO_DES_CBC, crypto_get_driverid(), CRYPTO_MD5, CRYPTO_MD5_HMAC, CRYPTO_MD5_KPDK, CRYPTO_NULL_CBC, CRYPTO_NULL_HMAC, crypto_register(), CRYPTO_RIJNDAEL128_CBC, CRYPTO_RIPEMD160_HMAC, CRYPTO_SHA1, CRYPTO_SHA1_HMAC, CRYPTO_SHA1_KPDK, CRYPTO_SHA2_256_HMAC, CRYPTO_SHA2_384_HMAC, CRYPTO_SHA2_512_HMAC, CRYPTO_SKIPJACK_CBC, CRYPTOCAP_F_SOFTWARE, CRYPTOCAP_F_SYNC, hmac_ipad_buffer, HMAC_IPAD_VAL, HMAC_MAX_BLOCK_LEN, hmac_opad_buffer, HMAC_OPAD_VAL, REGISTER, swcr_freesession(), swcr_id, swcr_newsession(), and swcr_process().

Here is the call graph for this function:



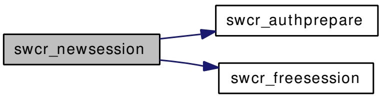
7.10.2.8 static int swcr_newsession (void ∗, u_int32_t ∗, struct cryptoini ∗) [static]

Definition at line 583 of file cryptosoft.c.

References auth_hash_hmac_md5, auth_hash_hmac_ripemd_160, auth_hash_hmac_sha1, auth_hash_hmac_sha2_256, auth_hash_hmac_sha2_384, auth_hash_hmac_sha2_512, auth_hash_key_md5, auth_hash_key_sha1, auth_hash_null, comp_algo_deflate, cryptoini::cri_alg, cryptoini::cri_key, cryptoini::cri_klen, cryptoini::cri_mlen, cryptoini::cri_next, CRYPTO_3DES_CBC, CRYPTO_BLF_CBC, CRYPTO_CAST_CBC, CRYPTO_DEFLATE_COMP, CRYPTO_DES_CBC, CRYPTO_MD5, CRYPTO_MD5_HMAC, CRYPTO_MD5_KPDK, CRYPTO_NULL_CBC, CRYPTO_NULL_HMAC, CRYPTO_RIJNDAEL128_CBC, CRYPTO_RIPEMD160_HMAC, CRYPTO_SHA1, CRYPTO_SHA1_HMAC, CRYPTO_SHA1_KPDK, CRYPTO_SHA2_256_HMAC, CRYPTO_SHA2_384_HMAC, CRYPTO_SHA2_512_HMAC, CRYPTO_SKIPJACK_CBC, CRYPTO_SW_SESSIONS, auth_hash::ctxsize, enc_xform_3des, enc_xform_blf, enc_xform_cast5, enc_xform_des, enc_xform_null, enc_xform_rijndael128, enc_xform_skipjack, auth_hash::Init, enc_xform::setkey, swcr_authprepare(), swcr_freesession(), swcr_sesnum, and swcr_sessions.

Referenced by swcr_init().

Here is the call graph for this function:



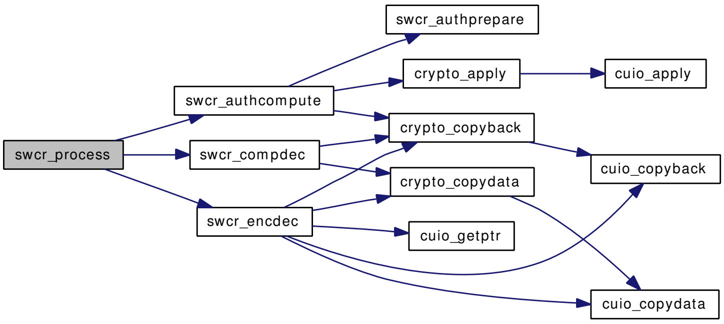
7.10.2.9 static int swcr_process (void ∗, struct cryptop ∗, int) [static]

Definition at line 877 of file cryptosoft.c.

References cryptodesc::crd_next, cryptop::crp_buf, cryptop::crp_desc, cryptop::crp_etype, cryptop::crp_flags, cryptop::crp_olen, cryptop::crp_sid, CRYPTO_3DES_CBC, CRYPTO_BLF_CBC, CRYPTO_CAST_CBC, CRYPTO_DEFLATE_COMP, CRYPTO_DES_CBC, CRYPTO_MD5, CRYPTO_MD5_HMAC, CRYPTO_MD5_KPDK, CRYPTO_NULL_CBC, CRYPTO_NULL_HMAC, CRYPTO_RIJNDAEL128_CBC, CRYPTO_RIPEMD160_HMAC, CRYPTO_SHA1, CRYPTO_SHA1_HMAC, CRYPTO_SHA1_KPDK, CRYPTO_SHA2_256_HMAC, CRYPTO_SHA2_384_HMAC, CRYPTO_SHA2_512_HMAC, CRYPTO_SKIPJACK_CBC, swcr_data::sw_alg, swcr_data::sw_next, swcr_authcompute(), swcr_compdec(), swcr_encdec(), swcr_sesnum, and swcr_sessions.

Referenced by swcr_init().

Here is the call graph for this function:



7.10.2.10 static void swcr_uninit (void) [static]

Definition at line 1015 of file cryptosoft.c.

References hmac_ipad_buffer, hmac_opad_buffer, and swcr_sessions.

7.10.2.11 SYSUNINIT (cryptosoft_uninit, SI_SUB_PSEUDO, SI_ORDER_ANY, swcr_uninit, NULL)

7.10.3 Variable Documentation

7.10.3.1 u_int8_t∗ hmac_ipad_buffer

Definition at line 48 of file cryptosoft.c.

Referenced by swcr_authprepare(), swcr_init(), and swcr_uninit().

7.10.3.2 u_int8_t* hmac_opad_buffer

Definition at line 49 of file cryptosoft.c.

Referenced by swcr_authprepare(), swcr_init(), and swcr_uninit().

7.10.3.3 int32_t swcr_id = -1

Definition at line 53 of file cryptosoft.c.

Referenced by swcr_init().

7.10.3.4 u_int32_t swcr_sesnum = 0

Definition at line 52 of file cryptosoft.c.

Referenced by swcr_freesession(), swcr_newsession(), and swcr_process().

7.10.3.5 struct swcr_data** swcr_sessions = NULL

Definition at line 51 of file cryptosoft.c.

Referenced by swcr_freesession(), swcr_newsession(), swcr_process(), and swcr_uninit().

7.11 /usr/src/sys/opencrypto/cryptosoft.h File Reference

This graph shows which files directly or indirectly include this file:

/usr/src/sys/opencrypto/cryptosoft.h
/usr/src/sys/opencrypto/cryptosoft.c

Data Structures

- struct swcr_data

Defines

- #define sw_ictx SWCR_UN.SWCR_AUTH.SW_ictx
- #define sw_octx SWCR_UN.SWCR_AUTH.SW_octx
- #define sw_klen SWCR_UN.SWCR_AUTH.SW_klen
- #define sw_mlen SWCR_UN.SWCR_AUTH.SW_mlen
- #define sw_axf SWCR_UN.SWCR_AUTH.SW_axf
- #define sw_kschedule SWCR_UN.SWCR_ENC.SW_kschedule
- #define sw_exf SWCR_UN.SWCR_ENC.SW_exf
- #define sw_size SWCR_UN.SWCR_COMP.SW_size
- #define sw_cxf SWCR_UN.SWCR_COMP.SW_cxf

Variables

- u_int8_t ∗ hmac_ipad_buffer
- u_int8_t ∗ hmac_opad_buffer

7.11.1 Define Documentation

7.11.1.1 #define sw_axf SWCR_UN.SWCR_AUTH.SW_axf

Definition at line 53 of file cryptosoft.h.

7.11.1.2 #define sw_cxf SWCR_UN.SWCR_COMP.SW_cxf

Definition at line 57 of file cryptosoft.h.

7.11.1.3 #define sw_exf SWCR_UN.SWCR_ENC.SW_exf

Definition at line 55 of file cryptosoft.h.

7.11.1.4 #define sw_ictx SWCR_UN.SWCR_AUTH.SW_ictx

Definition at line 49 of file cryptosoft.h.

7.11.1.5 #define sw_klen SWCR_UN.SWCR_AUTH.SW_klen

Definition at line 51 of file cryptosoft.h.

7.11.1.6 #define sw_kschedule SWCR_UN.SWCR_ENC.SW_kschedule

Definition at line 54 of file cryptosoft.h.

7.11.1.7 #define sw_mlen SWCR_UN.SWCR_AUTH.SW_mlen

Definition at line 52 of file cryptosoft.h.

7.11.1.8 #define sw_octx SWCR_UN.SWCR_AUTH.SW_octx

Definition at line 50 of file cryptosoft.h.

7.11.1.9 #define sw_size SWCR_UN.SWCR_COMP.SW_size

Definition at line 56 of file cryptosoft.h.

7.11.2 Variable Documentation

7.11.2.1 u_int8_t* hmac_ipad_buffer

Definition at line 48 of file cryptosoft.c.

Referenced by swcr_authprepare(), swcr_init(), and swcr_uninit().

7.11.2.2 u_int8_t* hmac_opad_buffer

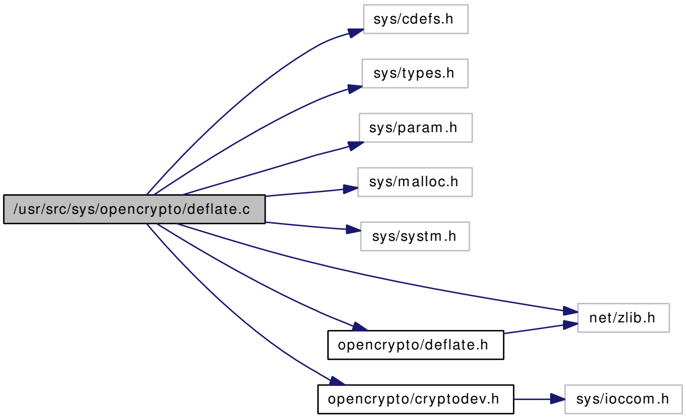
Definition at line 49 of file cryptosoft.c.

Referenced by swcr_authprepare(), swcr_init(), and swcr_uninit().

7.12 /usr/src/sys/opencrypto/deflate.c File Reference

```
#include <sys/cdefs.h>
#include <sys/types.h>
#include <sys/param.h>
#include <sys/malloc.h>
#include <sys/systm.h>
#include <net/zlib.h>
#include <opencrypto/cryptodev.h>
#include <opencrypto/deflate.h>
```

Include dependency graph for deflate.c:



Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/deflate.c,v 1.4 2005/05/30 05:01:44 scottl Exp $")
- u_int32_t deflate_global (u_int8_t ∗data, u_int32_t size, int decomp, u_int8_t ∗∗out)
- void ∗ z_alloc (void ∗nil, u_int type, u_int size)
- void z_free (void ∗nil, void ∗ptr)

Variables

- int window_inflate = -1 ∗ MAX_WBITS
- int window_deflate = -12

7.12.1 Function Documentation

7.12.1.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/deflate. *c*, v 1.4 2005/05/30 05:01:44 scottl Exp $")

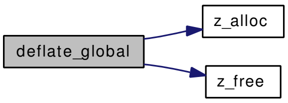
7.12.1.2 u_int32_t deflate_global (u_int8_t ∗ *data*, u_int32_t *size*, int *decomp*, u_int8_t ∗∗ *out*)

Definition at line 57 of file deflate.c.

References deflate_buf::flag, deflate_buf::out, deflate_buf::size, window_deflate, window_inflate, z_alloc(), z_free(), Z_MEMLEVEL, Z_METHOD, and ZBUF.

Referenced by deflate_compress(), and deflate_decompress().

Here is the call graph for this function:



7.12.1.3 void* z_alloc (void * *nil*, u_int *type*, u_int *size*)

Definition at line 177 of file deflate.c.

Referenced by deflate_global().

7.12.1.4 void z_free (void * *nil*, void * *ptr*)

Definition at line 188 of file deflate.c.

Referenced by deflate_global().

7.12.2 Variable Documentation

7.12.2.1 int window_deflate = -12

Definition at line 49 of file deflate.c.

Referenced by deflate_global().

7.12.2.2 int window_inflate = -1 * MAX_WBITS

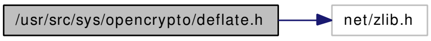
Definition at line 48 of file deflate.c.

Referenced by deflate_global().

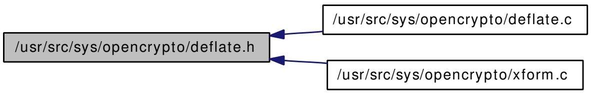
7.13 /usr/src/sys/opencrypto/deflate.h File Reference

```
#include <net/zlib.h>
```

Include dependency graph for deflate.h:



This graph shows which files directly or indirectly include this file:


/usr/src/sys/opencrypto/deflate.h
/usr/src/sys/opencrypto/deflate.c
/usr/src/sys/opencrypto/xform.c

Data Structures

- struct deflate_buf

Defines

- #define Z_METHOD 8
- #define Z_MEMLEVEL 8
- #define MINCOMP 2
- #define ZBUF 10

Functions

- u_int32_t deflate_global (u_int8_t ∗, u_int32_t, int, u_int8_t ∗∗)
- void ∗ z_alloc (void ∗, u_int, u_int)
- void z_free (void ∗, void ∗)

7.13.1 Define Documentation

7.13.1.1 #define MINCOMP 2

Definition at line 43 of file deflate.h.

7.13.1.2 #define Z_MEMLEVEL 8

Definition at line 42 of file deflate.h.

Referenced by deflate_global().

7.13.1.3 #define Z_METHOD 8

Definition at line 41 of file deflate.h.

Referenced by deflate_global().

7.13.1.4 #define ZBUF 10

Definition at line 44 of file deflate.h.

Referenced by deflate_global().

7.13.2 Function Documentation

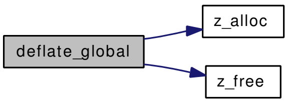
7.13.2.1 u_int32_t deflate_global (u_int8_t ∗, u_int32_t, int, u_int8_t ∗∗)

Definition at line 57 of file deflate.c.

References deflate_buf::flag, deflate_buf::out, deflate_buf::size, window_deflate, window_inflate, z_alloc(), z_free(), Z_MEMLEVEL, Z_METHOD, and ZBUF.

Referenced by deflate_compress(), and deflate_decompress().

Here is the call graph for this function:



7.13.2.2 void∗ z_alloc (void ∗, u_int, u_int)

Definition at line 177 of file deflate.c.

Referenced by deflate_global().

7.13.2.3 void z_free (void ∗, void ∗)

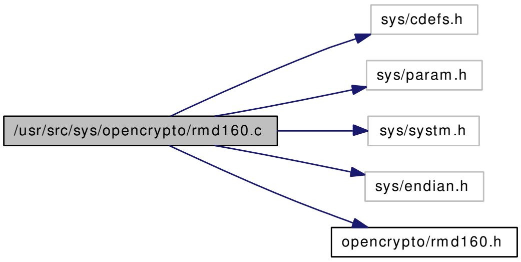
Definition at line 188 of file deflate.c.

Referenced by deflate_global().

7.14 /usr/src/sys/opencrypto/rmd160.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/systm.h>
#include <sys/endian.h>
#include <opencrypto/rmd160.h>
```

Include dependency graph for rmd160.c:



Defines

- #define PUT_64BIT_LE(cp, value)
- #define PUT_32BIT_LE(cp, value)
- #define H0 0x67452301U
- #define H1 0xEFCDAB89U
- #define H2 0x98BADCFEU
- #define H3 0x10325476U
- #define H4 0xC3D2E1F0U
- #define K0 0x00000000U
- #define K1 0x5A827999U
- #define K2 0x6ED9EBA1U
- #define K3 0x8F1BBCDCU
- #define K4 0xA953FD4EU
- #define KK0 0x50A28BE6U
- #define KK1 0x5C4DD124U
- #define KK2 0x6D703EF3U
- #define KK3 0x7A6D76E9U
- #define KK4 0x00000000U
- #define ROL(n, x) (((x) << (n)) | ((x) >> (32-(n))))
- #define F0(x, y, z) ((x) ^ (y) ^ (z))
- #define F1(x, y, z) (((x) & (y)) | ((∼x) & (z)))
- #define F2(x, y, z) (((x) | (∼y)) ^ (z))
- #define F3(x, y, z) (((x) & (z)) | ((y) & (∼z)))
- #define F4(x, y, z) ((x) ^ ((y) | (∼z)))
- #define R(a, b, c, d, e, Fj, Kj, sj, rj)
- #define X(i) x[i]

Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/rmd160.c,v 1.3 2005/01/07 02:29:16 imp Exp $")
- void RMD160Init (RMD160_CTX ∗ctx)
- void RMD160Update (RMD160_CTX ∗ctx, const u_char ∗input, u_int32_t len)
- void RMD160Final (u_char digest[20], RMD160_CTX ∗ctx)
- void RMD160Transform (u_int32_t state[5], const u_char block[64])

Variables

- static u_char PADDING [64]

7.14.1 Define Documentation

7.14.1.1 #define F0(x, y, z) ((x) ^ (y) ^ (z))

Definition at line 77 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.2 #define F1(x, y, z) (((x) & (y)) | ((∼x) & (z)))

Definition at line 78 of file rmd160.c.

7.14.1.3 #define F2(x, y, z) (((x) | (∼y)) ^ (z))

Definition at line 79 of file rmd160.c.

7.14.1.4 #define F3(x, y, z) (((x) & (z)) | ((y) & (∼z)))

Definition at line 80 of file rmd160.c.

7.14.1.5 #define F4(x, y, z) ((x) ^ ((y) | (∼z)))

Definition at line 81 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.6 #define H0 0x67452301U

Definition at line 56 of file rmd160.c.

Referenced by RMD160Init().

7.14.1.7 #define H1 0xEFCDAB89U

Definition at line 57 of file rmd160.c.

Referenced by RMD160Init().

7.14.1.8 #define H2 0x98BADCFEU

Definition at line 58 of file rmd160.c.

Referenced by RMD160Init().

7.14.1.9 #define H3 0x10325476U

Definition at line 59 of file rmd160.c.

Referenced by RMD160Init().

7.14.1.10 #define H4 0xC3D2E1F0U

Definition at line 60 of file rmd160.c.

Referenced by RMD160Init().

7.14.1.11 #define K0 0x00000000U

Definition at line 62 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.12 #define K1 0x5A827999U

Definition at line 63 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.13 #define K2 0x6ED9EBA1U

Definition at line 64 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.14 #define K3 0x8F1BBCDCU

Definition at line 65 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.15 #define K4 0xA953FD4EU

Definition at line 66 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.16 #define KK0 0x50A28BE6U

Definition at line 68 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.17 #define KK1 0x5C4DD124U

Definition at line 69 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.18 #define KK2 0x6D703EF3U

Definition at line 70 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.19 #define KK3 0x7A6D76E9U

Definition at line 71 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.20 #define KK4 0x00000000U

Definition at line 72 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.21 #define PUT_32BIT_LE(cp, value)

Value:

```
do { \
        (cp)[3] = (value) >> 24; \
        (cp)[2] = (value) >> 16; \
        (cp)[1] = (value) >> 8; \
        (cp)[0] = (value); } while (0)
```

Definition at line 50 of file rmd160.c.

Referenced by RMD160Final().

7.14.1.22 #define PUT_64BIT_LE(cp, value)

Value:

```
do { \
        (cp)[7] = (value) >> 56; \
        (cp)[6] = (value) >> 48; \
        (cp)[5] = (value) >> 40; \
        (cp)[4] = (value) >> 32; \
        (cp)[3] = (value) >> 24; \
        (cp)[2] = (value) >> 16; \
        (cp)[1] = (value) >> 8; \
        (cp)[0] = (value); } while (0)
```

Definition at line 40 of file rmd160.c.

Referenced by RMD160Final().

7.14.1.23 #define R(a, b, c, d, e, Fj, Kj, sj, rj)

Value:

```
do { \
                a = ROL(sj, a + Fj(b,c,d) + X(rj) + Kj) + e; \
                c = ROL(10, c); \
        } while(0)
```

Definition at line 83 of file rmd160.c.

Referenced by RMD160Transform().

7.14.1.24 #define ROL(n, x) (((x) << (n)) | ((x) >> (32-(n))))

Definition at line 75 of file rmd160.c.

7.14.1.25 #define X(i) x[i]

Definition at line 89 of file rmd160.c.

7.14.2 Function Documentation

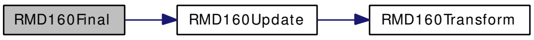
7.14.2.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/rmd160. *c*, v 1.3 2005/01/07 02:29:16 imp Exp $")

7.14.2.2 void RMD160Final (u_char *digest*[20], RMD160_CTX ∗ *ctx*)

Definition at line 136 of file rmd160.c.

References RMD160Context::count, PADDING, PUT_32BIT_LE, PUT_64BIT_LE, RMD160Update(), and RMD160Context::state.

Here is the call graph for this function:



7.14.2.3 void RMD160Init (RMD160_CTX ∗ *ctx*)

Definition at line 98 of file rmd160.c.

References RMD160Context::count, H0, H1, H2, H3, H4, and RMD160Context::state.

7.14.2.4 void RMD160Transform (u_int32_t *state*[5], const u_char *block*[64])

Definition at line 162 of file rmd160.c.

References F0, F1, F2, F3, F4, K0, K1, K2, K3, K4, KK0, KK1, KK2, KK3, KK4, and R.

Referenced by RMD160Update().

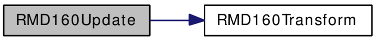
7.14.2.5 void RMD160Update (RMD160_CTX ∗ *ctx*, const u_char ∗ *input*, u_int32_t *len*)

Definition at line 109 of file rmd160.c.

References RMD160Context::buffer, RMD160Context::count, RMD160Transform(), and RMD160Context::state.

Referenced by RMD160Final(), and RMD160Update_int().

Here is the call graph for this function:



7.14.3 Variable Documentation

7.14.3.1 u_char PADDING[64] [static]

Initial value:

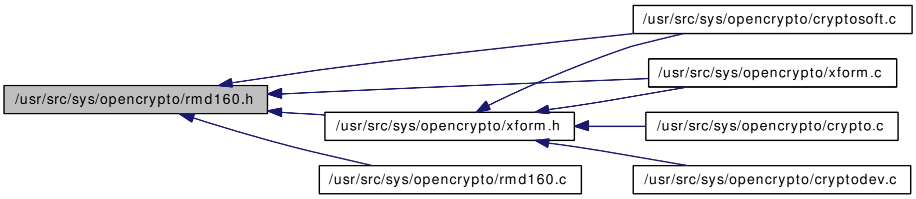
```
 {
        0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 91 of file rmd160.c.

Referenced by RMD160Final().

7.15 /usr/src/sys/opencrypto/rmd160.h File Reference

This graph shows which files directly or indirectly include this file:


/usr/src/sys/opencrypto/rmd160.h
/usr/src/sys/opencrypto/cryptosoft.c
/usr/src/sys/opencrypto/xform.c
/usr/src/sys/opencrypto/xform.h
/usr/src/sys/opencrypto/crypto.c
/usr/src/sys/opencrypto/rmd160.c
/usr/src/sys/opencrypto/cryptodev.c

Data Structures

- struct RMD160Context

Typedefs

- typedef RMD160Context RMD160_CTX

Functions

- void RMD160Init (RMD160_CTX ∗)
- void RMD160Transform (u_int32_t[5], const u_char[64])
- void RMD160Update (RMD160_CTX ∗, const u_char ∗, u_int32_t)
- void RMD160Final (u_char[20], RMD160_CTX ∗)

7.15.1 Typedef Documentation

7.15.1.1 typedef struct RMD160Context RMD160_CTX

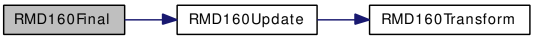
7.15.2 Function Documentation

7.15.2.1 void RMD160Final (u_char[20], RMD160_CTX ∗)

Definition at line 136 of file rmd160.c.

References RMD160Context::count, PADDING, PUT_32BIT_LE, PUT_64BIT_LE, RMD160Update(), and RMD160Context::state.

Here is the call graph for this function:



7.15.2.2 void RMD160Init (RMD160_CTX ∗)

Definition at line 98 of file rmd160.c.

References RMD160Context::count, H0, H1, H2, H3, H4, and RMD160Context::state.

7.15.2.3 void RMD160Transform (u_int32_t[5], const *u_char*[64])

Definition at line 162 of file rmd160.c.

References F0, F1, F2, F3, F4, K0, K1, K2, K3, K4, KK0, KK1, KK2, KK3, KK4, and R.

Referenced by RMD160Update().

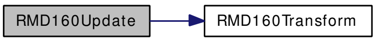
7.15.2.4 void RMD160Update (RMD160_CTX ∗, const u_char ∗, u_int32_t)

Definition at line 109 of file rmd160.c.

References RMD160Context::buffer, RMD160Context::count, RMD160Transform(), and RMD160Context::state.

Referenced by RMD160Final(), and RMD160Update_int().

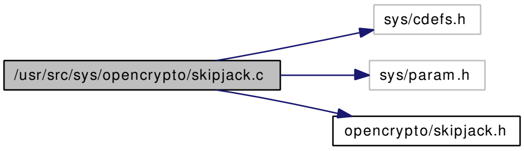
Here is the call graph for this function:



7.16 /usr/src/sys/opencrypto/skipjack.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <opencrypto/skipjack.h>
```

Include dependency graph for skipjack.c:



Defines

- #define g(k0, k1, k2, k3, ih, il, oh, ol)
- #define g0(ih, il, oh, ol) g(0, 1, 2, 3, ih, il, oh, ol)
- #define g4(ih, il, oh, ol) g(4, 5, 6, 7, ih, il, oh, ol)
- #define g8(ih, il, oh, ol) g(8, 9, 0, 1, ih, il, oh, ol)
- #define g2(ih, il, oh, ol) g(2, 3, 4, 5, ih, il, oh, ol)
- #define g6(ih, il, oh, ol) g(6, 7, 8, 9, ih, il, oh, ol)
- #define g_inv(k0, k1, k2, k3, ih, il, oh, ol)
- #define g0_inv(ih, il, oh, ol) g_inv(0, 1, 2, 3, ih, il, oh, ol)
- #define g4_inv(ih, il, oh, ol) g_inv(4, 5, 6, 7, ih, il, oh, ol)
- #define g8_inv(ih, il, oh, ol) g_inv(8, 9, 0, 1, ih, il, oh, ol)
- #define g2_inv(ih, il, oh, ol) g_inv(2, 3, 4, 5, ih, il, oh, ol)
- #define g6_inv(ih, il, oh, ol) g_inv(6, 7, 8, 9, ih, il, oh, ol)

Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/skipjack.c,v 1.3 2005/01/07 02:29:16 imp Exp $")
- void subkey_table_gen (u_int8_t ∗key, u_int8_t ∗∗key_tables)
- void skipjack_forwards (u_int8_t ∗plain, u_int8_t ∗cipher, u_int8_t ∗∗key_tables)
- void skipjack_backwards (u_int8_t ∗cipher, u_int8_t ∗plain, u_int8_t ∗∗key_tables)

Variables

- static const u_int8_t ftable [0x100]

7.16.1 Define Documentation

7.16.1.1 #define g(k0, k1, k2, k3, ih, il, oh, ol)

Value:

```
{ \
        oh = k##k0 [il] ^ ih; \
        ol = k##k1 [oh] ^ il; \
        oh = k##k2 [ol] ^ oh; \
        ol = k##k3 [oh] ^ ol; \
}
```

Definition at line 81 of file skipjack.c.

7.16.1.2 #define g0(ih, il, oh, ol) g(0, 1, 2, 3, ih, il, oh, ol)

Definition at line 89 of file skipjack.c.

Referenced by skipjack_forwards().

7.16.1.3 #define g0_inv(ih, il, oh, ol) g_inv(0, 1, 2, 3, ih, il, oh, ol)

Definition at line 105 of file skipjack.c.

Referenced by skipjack_backwards().

7.16.1.4 #define g2(ih, il, oh, ol) g(2, 3, 4, 5, ih, il, oh, ol)

Definition at line 92 of file skipjack.c.

Referenced by skipjack_forwards().

7.16.1.5 #define g2_inv(ih, il, oh, ol) g_inv(2, 3, 4, 5, ih, il, oh, ol)

Definition at line 108 of file skipjack.c.

Referenced by skipjack_backwards().

7.16.1.6 #define g4(ih, il, oh, ol) g(4, 5, 6, 7, ih, il, oh, ol)

Definition at line 90 of file skipjack.c.

Referenced by skipjack_forwards().

7.16.1.7 #define g4_inv(ih, il, oh, ol) g_inv(4, 5, 6, 7, ih, il, oh, ol)

Definition at line 106 of file skipjack.c.

Referenced by skipjack_backwards().

7.16.1.8 #define g6(ih, il, oh, ol) g(6, 7, 8, 9, ih, il, oh, ol)

Definition at line 93 of file skipjack.c.

Referenced by skipjack_forwards().

7.16.1.9 #define g6_inv(ih, il, oh, ol) g_inv(6, 7, 8, 9, ih, il, oh, ol)

Definition at line 109 of file skipjack.c.

Referenced by skipjack_backwards().

7.16.1.10 #define g8(ih, il, oh, ol) g(8, 9, 0, 1, ih, il, oh, ol)

Definition at line 91 of file skipjack.c.

Referenced by skipjack_forwards().

7.16.1.11 #define g8_inv(ih, il, oh, ol) g_inv(8, 9, 0, 1, ih, il, oh, ol)

Definition at line 107 of file skipjack.c.

Referenced by skipjack_backwards().

7.16.1.12 #define g_inv(k0, k1, k2, k3, ih, il, oh, ol)

Value:

```
{ \
        ol = k##k3 [ih] ^ il; \
        oh = k##k2 [ol] ^ ih; \
        ol = k##k1 [oh] ^ ol; \
        oh = k##k0 [ol] ^ oh; \
}
```

Definition at line 96 of file skipjack.c.

7.16.2 Function Documentation

7.16.2.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/skipjack. *c*, v 1.3 2005/01/07 02:29:16 imp Exp $")

7.16.2.2 void skipjack_backwards (u_int8_t ∗ *cipher*, u_int8_t ∗ *plain*, u_int8_t ∗∗ *key_tables*)

Definition at line 196 of file skipjack.c.

References g0_inv, g2_inv, g4_inv, g6_inv, and g8_inv.

Referenced by skipjack_decrypt().

7.16.2.3 void skipjack_forwards (u_int8_t ∗ *plain*, u_int8_t ∗ *cipher*, u_int8_t ∗∗ *key_tables*)

Definition at line 129 of file skipjack.c.

References g0, g2, g4, g6, and g8.

Referenced by skipjack_encrypt().

7.16.2.4 void subkey_table_gen (u_int8_t ∗ *key*, u_int8_t ∗∗ *key_tables*)

Definition at line 68 of file skipjack.c.

References ftable.

Referenced by skipjack_setkey().

7.16.3 Variable Documentation

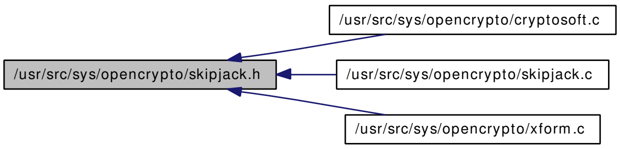
7.16.3.1 const u_int8_t ftable[0x100] `[static]`

Definition at line 22 of file skipjack.c.

Referenced by subkey_table_gen().

7.17 /usr/src/sys/opencrypto/skipjack.h File Reference

This graph shows which files directly or indirectly include this file:


/usr/src/sys/opencrypto/cryptosoft.c
/usr/src/sys/opencrypto/skipjack.h
/usr/src/sys/opencrypto/skipjack.c
/usr/src/sys/opencrypto/xform.c

Functions

- void skipjack_forwards (u_int8_t ∗plain, u_int8_t ∗cipher, u_int8_t ∗∗key)
- void skipjack_backwards (u_int8_t ∗cipher, u_int8_t ∗plain, u_int8_t ∗∗key)
- void subkey_table_gen (u_int8_t ∗key, u_int8_t ∗∗key_tables)

7.17.1 Function Documentation

7.17.1.1 void skipjack_backwards (u_int8_t ∗ *cipher*, u_int8_t ∗ *plain*, u_int8_t ∗∗ *key*)

Definition at line 196 of file skipjack.c.

References g0_inv, g2_inv, g4_inv, g6_inv, and g8_inv.

Referenced by skipjack_decrypt().

7.17.1.2 void skipjack_forwards (u_int8_t ∗ *plain*, u_int8_t ∗ *cipher*, u_int8_t ∗∗ *key*)

Definition at line 129 of file skipjack.c.

References g0, g2, g4, g6, and g8.

Referenced by skipjack_encrypt().

7.17.1.3 void subkey_table_gen (u_int8_t ∗ *key*, u_int8_t ∗∗ *key_tables*)

Definition at line 68 of file skipjack.c.

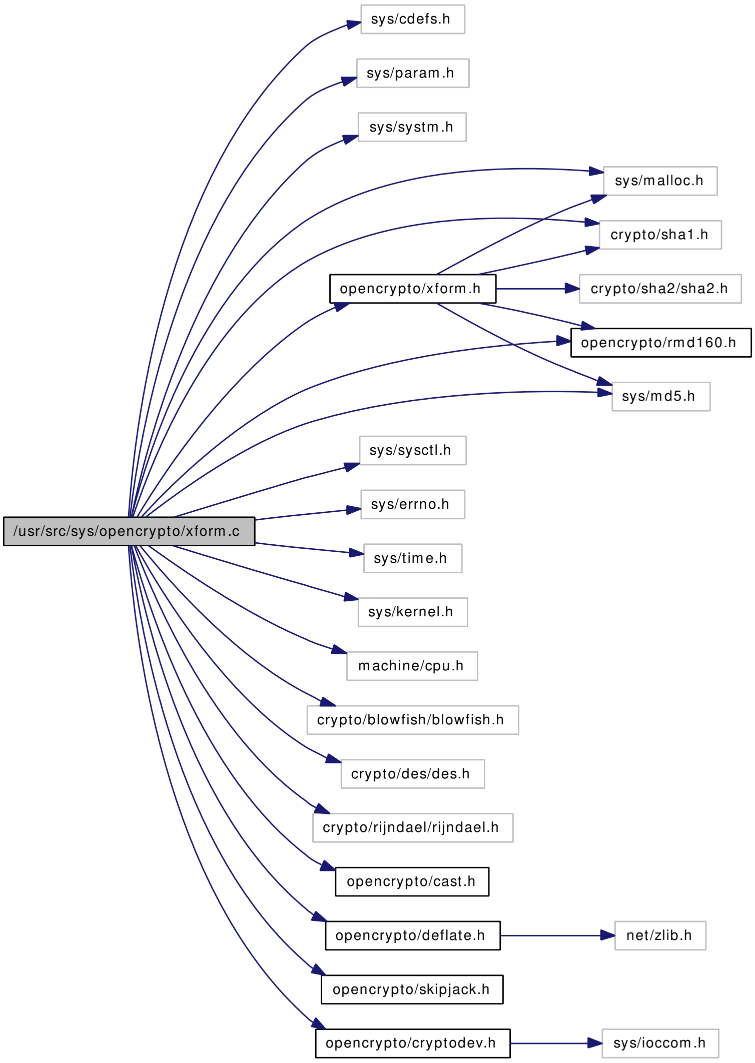
References ftable.

Referenced by skipjack_setkey().

7.18 /usr/src/sys/opencrypto/xform.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/systm.h>
#include <sys/malloc.h>
#include <sys/sysctl.h>
#include <sys/errno.h>
#include <sys/time.h>
#include <sys/kernel.h>
#include <machine/cpu.h>
#include <crypto/blowfish/blowfish.h>
#include <crypto/des/des.h>
#include <crypto/rijndael/rijndael.h>
#include <crypto/sha1.h>
#include <opencrypto/cast.h>
#include <opencrypto/deflate.h>
#include <opencrypto/rmd160.h>
#include <opencrypto/skipjack.h>
#include <sys/md5.h>
#include <opencrypto/cryptodev.h>
#include <opencrypto/xform.h>
```

Include dependency graph for xform.c:



Functions

- __FBSDID ("$FreeBSD: src/sys/opencrypto/xform.c,v 1.8 2006/06/04 15:11:59 pjd Exp $")
- static void null_encrypt (caddr_t, u_int8_t ∗)
- static void null_decrypt (caddr_t, u_int8_t ∗)
- static int null_setkey (u_int8_t ∗∗, u_int8_t ∗, int)
- static void null_zerokey (u_int8_t ∗∗)
- static int des1_setkey (u_int8_t ∗∗, u_int8_t ∗, int)
- static int des3_setkey (u_int8_t ∗∗, u_int8_t ∗, int)
- static int blf_setkey (u_int8_t ∗∗, u_int8_t ∗, int)
- static int cast5_setkey (u_int8_t ∗∗, u_int8_t ∗, int)
- static int skipjack_setkey (u_int8_t ∗∗, u_int8_t ∗, int)
- static int rijndael128_setkey (u_int8_t ∗∗, u_int8_t ∗, int)
- static void des1_encrypt (caddr_t, u_int8_t ∗)
- static void des3_encrypt (caddr_t, u_int8_t ∗)
- static void blf_encrypt (caddr_t, u_int8_t ∗)
- static void cast5_encrypt (caddr_t, u_int8_t ∗)

- static void skipjack_encrypt (caddr_t, u_int8_t ∗)
- static void rijndael128_encrypt (caddr_t, u_int8_t ∗)
- static void des1_decrypt (caddr_t, u_int8_t ∗)
- static void des3_decrypt (caddr_t, u_int8_t ∗)
- static void blf_decrypt (caddr_t, u_int8_t ∗)
- static void cast5_decrypt (caddr_t, u_int8_t ∗)
- static void skipjack_decrypt (caddr_t, u_int8_t ∗)
- static void rijndael128_decrypt (caddr_t, u_int8_t ∗)
- static void des1_zerokey (u_int8_t ∗∗)
- static void des3_zerokey (u_int8_t ∗∗)
- static void blf_zerokey (u_int8_t ∗∗)
- static void cast5_zerokey (u_int8_t ∗∗)
- static void skipjack_zerokey (u_int8_t ∗∗)
- static void rijndael128_zerokey (u_int8_t ∗∗)
- static void null_init (void ∗)
- static int null_update (void ∗, u_int8_t ∗, u_int16_t)
- static void null_final (u_int8_t ∗, void ∗)
- static int MD5Update_int (void ∗, u_int8_t ∗, u_int16_t)
- static void SHA1Init_int (void ∗)
- static int SHA1Update_int (void ∗, u_int8_t ∗, u_int16_t)
- static void SHA1Final_int (u_int8_t ∗, void ∗)
- static int RMD160Update_int (void ∗, u_int8_t ∗, u_int16_t)
- static int SHA256Update_int (void ∗, u_int8_t ∗, u_int16_t)
- static int SHA384Update_int (void ∗, u_int8_t ∗, u_int16_t)
- static int SHA512Update_int (void ∗, u_int8_t ∗, u_int16_t)
- static u_int32_t deflate_compress (u_int8_t ∗, u_int32_t, u_int8_t ∗∗)
- static u_int32_t deflate_decompress (u_int8_t ∗, u_int32_t, u_int8_t ∗∗)
- MALLOC_DEFINE (M_XDATA,"xform","xform data buffers")

Variables

- enc_xform enc_xform_null
- enc_xform enc_xform_des
- enc_xform enc_xform_3des
- enc_xform enc_xform_blf
- enc_xform enc_xform_cast5
- enc_xform enc_xform_skipjack
- enc_xform enc_xform_rijndael128
- enc_xform enc_xform_arc4
- auth_hash auth_hash_null
- auth_hash auth_hash_hmac_md5
- auth_hash auth_hash_hmac_sha1
- auth_hash auth_hash_hmac_ripemd_160
- auth_hash auth_hash_key_md5
- auth_hash auth_hash_key_sha1
- auth_hash auth_hash_hmac_sha2_256
- auth_hash auth_hash_hmac_sha2_384
- auth_hash auth_hash_hmac_sha2_512
- comp_algo comp_algo_deflate

7.18.1 Function Documentation

7.18.1.1 __FBSDID ("$FreeBSD: src/sys/opencrypto/xform. *c*, v 1.8 2006/06/04 15:11:59 pjd Exp $")

7.18.1.2 static void blf_decrypt (caddr_t, u_int8_t ∗) `[static]`

Definition at line 382 of file xform.c.

7.18.1.3 static void blf_encrypt (caddr_t, u_int8_t ∗) `[static]`

Definition at line 367 of file xform.c.

7.18.1.4 static int blf_setkey (u_int8_t ∗∗, u_int8_t ∗, int) `[static]`

Definition at line 397 of file xform.c.

7.18.1.5 static void blf_zerokey (u_int8_t ∗∗) `[static]`

Definition at line 412 of file xform.c.

7.18.1.6 static void cast5_decrypt (caddr_t, u_int8_t ∗) `[static]`

Definition at line 426 of file xform.c.

References cast_decrypt().

Here is the call graph for this function:

7.18.1.7 static void cast5_encrypt (caddr_t, u_int8_t ∗) `[static]`

Definition at line 420 of file xform.c.

References cast_encrypt().

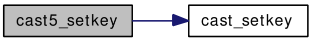
Here is the call graph for this function:

7.18.1.8 static int cast5_setkey (u_int8_t ∗∗, u_int8_t ∗, int) `[static]`

Definition at line 432 of file xform.c.

References cast_setkey().

Here is the call graph for this function:



7.18.1.9 static void cast5_zerokey (u_int8_t **) [static]

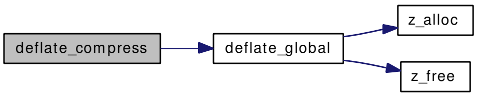
Definition at line 446 of file xform.c.

7.18.1.10 static u_int32_t deflate_compress (u_int8_t *, u_int32_t, u_int8_t **) [static]

Definition at line 617 of file xform.c.

References deflate_global().

Here is the call graph for this function:

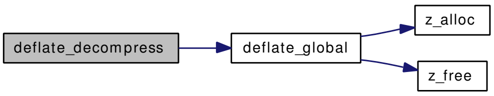


7.18.1.11 static u_int32_t deflate_decompress (u_int8_t *, u_int32_t, u_int8_t **) [static]

Definition at line 626 of file xform.c.

References deflate_global().

Here is the call graph for this function:



7.18.1.12 static void des1_decrypt (caddr_t, u_int8_t *) [static]

Definition at line 288 of file xform.c.

7.18.1.13 static void des1_encrypt (caddr_t, u_int8_t *) [static]

Definition at line 279 of file xform.c.

7.18.1.14 static int des1_setkey (u_int8_t **, u_int8_t *, int) [static]

Definition at line 297 of file xform.c.

7.18.1.15 static void des1_zerokey (u_int8_t **) [static]

Definition at line 314 of file xform.c.

7.18.1.16 static void des3_decrypt (caddr_t, u_int8_t ∗) [static]

Definition at line 331 of file xform.c.

7.18.1.17 static void des3_encrypt (caddr_t, u_int8_t ∗) [static]

Definition at line 322 of file xform.c.

7.18.1.18 static int des3_setkey (u_int8_t ∗∗, u_int8_t ∗, int) [static]

Definition at line 340 of file xform.c.

7.18.1.19 static void des3_zerokey (u_int8_t ∗∗) [static]

Definition at line 359 of file xform.c.

7.18.1.20 MALLOC_DEFINE (M_XDATA, "xform", "xform data buffers")

7.18.1.21 static int MD5Update_int (void ∗, u_int8_t ∗, u_int16_t) [static]

Definition at line 566 of file xform.c.

7.18.1.22 static void null_decrypt (caddr_t, u_int8_t ∗) [static]

Definition at line 263 of file xform.c.

7.18.1.23 static void null_encrypt (caddr_t, u_int8_t ∗) [static]

Definition at line 259 of file xform.c.

7.18.1.24 static void null_final (u_int8_t ∗, void ∗) [static]

Definition at line 552 of file xform.c.

7.18.1.25 static void null_init (void ∗) [static]

Definition at line 541 of file xform.c.

7.18.1.26 static int null_setkey (u_int8_t ∗∗, u_int8_t ∗, int) [static]

Definition at line 267 of file xform.c.

7.18.1.27 static int null_update (void ∗, u_int8_t ∗, u_int16_t) [static]

Definition at line 546 of file xform.c.

7.18.1.28 static void null_zerokey (u_int8_t ∗∗) [static]

Definition at line 273 of file xform.c.

7.18.1.29 static void rijndael128_decrypt (caddr_t, u_int8_t ∗) [static]

Definition at line 504 of file xform.c.

7.18.1.30 static void rijndael128_encrypt (caddr_t, u_int8_t ∗) [static]

Definition at line 498 of file xform.c.

7.18.1.31 static int rijndael128_setkey (u_int8_t ∗∗, u_int8_t ∗, int) [static]

Definition at line 511 of file xform.c.

7.18.1.32 static void rijndael128_zerokey (u_int8_t ∗∗) [static]

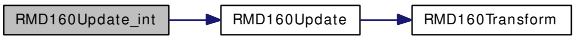
Definition at line 529 of file xform.c.

7.18.1.33 static int RMD160Update_int (void ∗, u_int8_t ∗, u_int16_t) [static]

Definition at line 559 of file xform.c.

References RMD160Update().

Here is the call graph for this function:



7.18.1.34 static void SHA1Final_int (u_int8_t ∗, void ∗) [static]

Definition at line 586 of file xform.c.

7.18.1.35 static void SHA1Init_int (void ∗) [static]

Definition at line 573 of file xform.c.

7.18.1.36 static int SHA1Update_int (void ∗, u_int8_t ∗, u_int16_t) [static]

Definition at line 579 of file xform.c.

7.18.1.37 static int SHA256Update_int (void ∗, u_int8_t ∗, u_int16_t) [static]

Definition at line 592 of file xform.c.

7.18.1.38 static int SHA384Update_int (void *, u_int8_t *, u_int16_t) [static]

Definition at line 599 of file xform.c.

7.18.1.39 static int SHA512Update_int (void *, u_int8_t *, u_int16_t) [static]

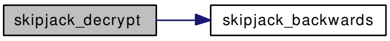
Definition at line 606 of file xform.c.

7.18.1.40 static void skipjack_decrypt (caddr_t, u_int8_t *) [static]

Definition at line 460 of file xform.c.

References skipjack_backwards().

Here is the call graph for this function:

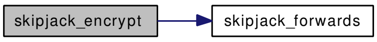


7.18.1.41 static void skipjack_encrypt (caddr_t, u_int8_t *) [static]

Definition at line 454 of file xform.c.

References skipjack_forwards().

Here is the call graph for this function:

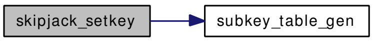


7.18.1.42 static int skipjack_setkey (u_int8_t **, u_int8_t *, int) [static]

Definition at line 466 of file xform.c.

References subkey_table_gen().

Here is the call graph for this function:



7.18.1.43 static void skipjack_zerokey (u_int8_t **) [static]

Definition at line 490 of file xform.c.

7.18.2 Variable Documentation

7.18.2.1 struct auth_hash auth_hash_hmac_md5

Initial value:

```
 {
        CRYPTO_MD5_HMAC, "HMAC-MD5",
        16, MD5_HASH_LEN, MD5_HMAC_BLOCK_LEN, sizeof(MD5_CTX),
        (void (*) (void *)) MD5Init, MD5Update_int,
        (void (*) (u_int8_t *, void *)) MD5Final
}
```

Definition at line 194 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.2 struct auth_hash auth_hash_hmac_ripemd_160

Initial value:

```
 {
        CRYPTO_RIPEMD160_HMAC, "HMAC-RIPEMD-160",
        20, RIPEMD160_HASH_LEN, RIPEMD160_HMAC_BLOCK_LEN, sizeof(RMD160_CTX),
        (void (*)(void *)) RMD160Init, RMD160Update_int,
        (void (*)(u_int8_t *, void *)) RMD160Final
}
```

Definition at line 207 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.3 struct auth_hash auth_hash_hmac_sha1

Initial value:

```
 {
        CRYPTO_SHA1_HMAC, "HMAC-SHA1",
        20, SHA1_HASH_LEN, SHA1_HMAC_BLOCK_LEN, sizeof(SHA1_CTX),
        SHA1Init_int, SHA1Update_int, SHA1Final_int
}
```

Definition at line 201 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.4 struct auth_hash auth_hash_hmac_sha2_256

Initial value:

```
 {
        CRYPTO_SHA2_256_HMAC, "HMAC-SHA2-256",
        32, SHA2_256_HASH_LEN, SHA2_256_HMAC_BLOCK_LEN, sizeof(SHA256_CTX),
        (void (*)(void *)) SHA256_Init, SHA256Update_int,
        (void (*)(u_int8_t *, void *)) SHA256_Final
}
```

Definition at line 227 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.5 struct auth_hash auth_hash_hmac_sha2_384

Initial value:

```
 {
        CRYPTO_SHA2_384_HMAC, "HMAC-SHA2-384",
        48, SHA2_384_HASH_LEN, SHA2_384_HMAC_BLOCK_LEN, sizeof(SHA384_CTX),
        (void (*)(void *)) SHA384_Init, SHA384Update_int,
        (void (*)(u_int8_t *, void *)) SHA384_Final
}
```

Definition at line 234 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.6 struct auth_hash auth_hash_hmac_sha2_512

Initial value:

```
 {
        CRYPTO_SHA2_512_HMAC, "HMAC-SHA2-512",
        64, SHA2_512_HASH_LEN, SHA2_512_HMAC_BLOCK_LEN, sizeof(SHA512_CTX),
        (void (*)(void *)) SHA512_Init, SHA512Update_int,
        (void (*)(u_int8_t *, void *)) SHA512_Final
}
```

Definition at line 241 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.7 struct auth_hash auth_hash_key_md5

Initial value:

```
 {
        CRYPTO_MD5_KPDK, "Keyed MD5",
        0, MD5_KPDK_HASH_LEN, 0, sizeof(MD5_CTX),
        (void (*)(void *)) MD5Init, MD5Update_int,
        (void (*)(u_int8_t *, void *)) MD5Final
}
```

Definition at line 214 of file xform.c.

Referenced by swcr_newsession().

7.18.2.8 struct auth_hash auth_hash_key_sha1

Initial value:

```
 {
        CRYPTO_SHA1_KPDK, "Keyed SHA1",
        0, SHA1_KPDK_HASH_LEN, 0, sizeof(SHA1_CTX),
        SHA1Init_int, SHA1Update_int, SHA1Final_int
}
```

Definition at line 221 of file xform.c.

Referenced by swcr_newsession().

7.18.2.9 struct auth_hash auth_hash_null

Initial value:

```
 {
        CRYPTO_NULL_HMAC, "NULL-HMAC",
        0, NULL_HASH_LEN, NULL_HMAC_BLOCK_LEN, sizeof(int),
        null_init, null_update, null_final
}
```

Definition at line 188 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.10 struct comp_algo comp_algo_deflate

Initial value:

```
 {
        CRYPTO_DEFLATE_COMP, "Deflate",
        90, deflate_compress,
        deflate_decompress
}
```

Definition at line 249 of file xform.c.

Referenced by swcr_newsession().

7.18.2.11 struct enc_xform enc_xform_3des

Initial value:

```
 {
        CRYPTO_3DES_CBC, "3DES",
        DES3_BLOCK_LEN, 24, 24,
        des3_encrypt,
        des3_decrypt,
        des3_setkey,
        des3_zerokey
}
```

Definition at line 133 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.12 struct enc_xform enc_xform_arc4

Initial value:

```
 {
        CRYPTO_ARC4, "ARC4",
        1, 1, 32,
        NULL,
        NULL,
        NULL,
        NULL,
}
```

Definition at line 178 of file xform.c.

Referenced by cryptof_ioctl().

7.18.2.13 struct enc_xform enc_xform_blf

Initial value:

```
 {
        CRYPTO_BLF_CBC, "Blowfish",
        BLOWFISH_BLOCK_LEN, 5, 56 ,
        blf_encrypt,
        blf_decrypt,
        blf_setkey,
        blf_zerokey
}
```

Definition at line 142 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.14 struct enc_xform enc_xform_cast5

Initial value:

```
 {
        CRYPTO_CAST_CBC, "CAST-128",
        CAST128_BLOCK_LEN, 5, 16,
        cast5_encrypt,
        cast5_decrypt,
        cast5_setkey,
        cast5_zerokey
}
```

Definition at line 151 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.15 struct enc_xform enc_xform_des

Initial value:

```
 {
        CRYPTO_DES_CBC, "DES",
        DES_BLOCK_LEN, 8, 8,
        des1_encrypt,
        des1_decrypt,
        des1_setkey,
        des1_zerokey,
}
```

Definition at line 124 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.16 struct enc_xform enc_xform_null

Initial value:

```
 {
        CRYPTO_NULL_CBC, "NULL",

        NULL_BLOCK_LEN, 0, 256,
        null_encrypt,
        null_decrypt,
        null_setkey,
        null_zerokey,
}
```

Definition at line 114 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.17 struct enc_xform enc_xform_rijndael128

Initial value:

```
 {
        CRYPTO_RIJNDAEL128_CBC, "Rijndael-128/AES",
        RIJNDAEL128_BLOCK_LEN, 8, 32,
        rijndael128_encrypt,
        rijndael128_decrypt,
        rijndael128_setkey,
        rijndael128_zerokey,
}
```

Definition at line 169 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.18.2.18 struct enc_xform enc_xform_skipjack

Initial value:

```
 {
        CRYPTO_SKIPJACK_CBC, "Skipjack",
        SKIPJACK_BLOCK_LEN, 10, 10,
        skipjack_encrypt,
        skipjack_decrypt,
        skipjack_setkey,
        skipjack_zerokey
}
```

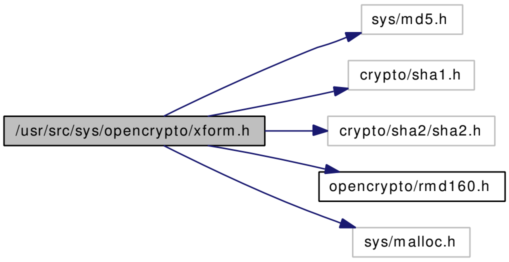
Definition at line 160 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

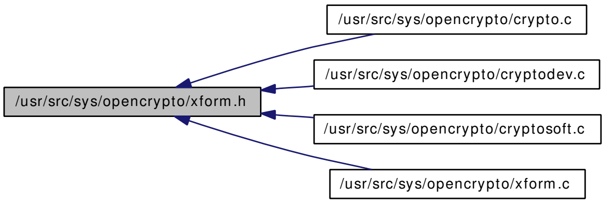
7.19 /usr/src/sys/opencrypto/xform.h File Reference

```
#include <sys/md5.h>
#include <crypto/sha1.h>
#include <crypto/sha2/sha2.h>
#include <opencrypto/rmd160.h>
#include <sys/malloc.h>
```

Include dependency graph for xform.h:



This graph shows which files directly or indirectly include this file:


/usr/src/sys/opencrypto/xform.h
/usr/src/sys/opencrypto/crypto.c
/usr/src/sys/opencrypto/cryptodev.c
/usr/src/sys/opencrypto/cryptosoft.c
/usr/src/sys/opencrypto/xform.c

Data Structures

- struct auth_hash
- struct enc_xform
- struct comp_algo
- union authctx

Defines

- #define AH_ALEN_MAX 20

Functions

- MALLOC_DECLARE (M_XDATA)

Variables

- enc_xform enc_xform_null

- enc_xform enc_xform_des
- enc_xform enc_xform_3des
- enc_xform enc_xform_blf
- enc_xform enc_xform_cast5
- enc_xform enc_xform_skipjack
- enc_xform enc_xform_rijndael128
- enc_xform enc_xform_arc4
- auth_hash auth_hash_null
- auth_hash auth_hash_key_md5
- auth_hash auth_hash_key_sha1
- auth_hash auth_hash_hmac_md5
- auth_hash auth_hash_hmac_sha1
- auth_hash auth_hash_hmac_ripemd_160
- auth_hash auth_hash_hmac_sha2_256
- auth_hash auth_hash_hmac_sha2_384
- auth_hash auth_hash_hmac_sha2_512
- comp_algo comp_algo_deflate

7.19.1 Define Documentation

7.19.1.1 #define AH_ALEN_MAX 20

Definition at line 46 of file xform.h.

7.19.2 Function Documentation

7.19.2.1 MALLOC_DECLARE (M_XDATA)

7.19.3 Variable Documentation

7.19.3.1 struct auth_hash auth_hash_hmac_md5

Definition at line 194 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.2 struct auth_hash auth_hash_hmac_ripemd_160

Definition at line 207 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.3 struct auth_hash auth_hash_hmac_sha1

Definition at line 201 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.4 struct auth_hash auth_hash_hmac_sha2_256

Definition at line 227 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.5 struct auth_hash auth_hash_hmac_sha2_384

Definition at line 234 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.6 struct auth_hash auth_hash_hmac_sha2_512

Definition at line 241 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.7 struct auth_hash auth_hash_key_md5

Definition at line 214 of file xform.c.

Referenced by swcr_newsession().

7.19.3.8 struct auth_hash auth_hash_key_sha1

Definition at line 221 of file xform.c.

Referenced by swcr_newsession().

7.19.3.9 struct auth_hash auth_hash_null

Definition at line 188 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.10 struct comp_algo comp_algo_deflate

Definition at line 249 of file xform.c.

Referenced by swcr_newsession().

7.19.3.11 struct enc_xform enc_xform_3des

Definition at line 133 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.12 struct enc_xform enc_xform_arc4

Definition at line 178 of file xform.c.

Referenced by cryptof_ioctl().

7.19.3.13 struct enc_xform enc_xform_blf

Definition at line 142 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.14 struct enc_xform enc_xform_cast5

Definition at line 151 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.15 struct enc_xform enc_xform_des

Definition at line 124 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.16 struct enc_xform enc_xform_null

Definition at line 114 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.17 struct enc_xform enc_xform_rijndael128

Definition at line 169 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

7.19.3.18 struct enc_xform enc_xform_skipjack

Definition at line 160 of file xform.c.

Referenced by cryptof_ioctl(), and swcr_newsession().

Index

/usr/ Directory Reference, 12
/usr/src/ Directory Reference, 10
/usr/src/sys/ Directory Reference, 11
/usr/src/sys/opencrypto/ Directory Reference, 9
/usr/src/sys/opencrypto/cast.c, 52
/usr/src/sys/opencrypto/cast.h, 55
/usr/src/sys/opencrypto/castsb.h, 56
/usr/src/sys/opencrypto/criov.c, 58
/usr/src/sys/opencrypto/crypto.c, 61
/usr/src/sys/opencrypto/crypto_if.m, 73
/usr/src/sys/opencrypto/cryptodev.c, 74
/usr/src/sys/opencrypto/cryptodev.h, 82
/usr/src/sys/opencrypto/cryptosoft.c, 103
/usr/src/sys/opencrypto/cryptosoft.h, 110
/usr/src/sys/opencrypto/deflate.c, 112
/usr/src/sys/opencrypto/deflate.h, 114
/usr/src/sys/opencrypto/rmd160.c, 116
/usr/src/sys/opencrypto/rmd160.h, 122
/usr/src/sys/opencrypto/skipjack.c, 124
/usr/src/sys/opencrypto/skipjack.h, 128
/usr/src/sys/opencrypto/xform.c, 129
/usr/src/sys/opencrypto/xform.h, 142
__FBSDID
    cast.c, 54
    criov.c, 59
    crypto.c, 64
    cryptodev.c, 76
    cryptosoft.c, 105
    deflate.c, 112
    rmd160.c, 120
    skipjack.c, 126
    xform.c, 132

acc
    cryptotstat, 39
AES_BLOCK_LEN
    cryptodev.h, 85
AH_ALEN_MAX
    xform.h, 143
auth_hash, 13
    blocksize, 13
    ctxsize, 13
    Final, 14
    hashsize, 14
    Init, 14
    keysize, 14
    name, 14
    type, 14
    Update, 14
auth_hash_hmac_md5
    xform.c, 137
    xform.h, 143
auth_hash_hmac_ripemd_160
    xform.c, 137
    xform.h, 143
auth_hash_hmac_sha1
    xform.c, 137
    xform.h, 143
auth_hash_hmac_sha2_256
    xform.c, 137
    xform.h, 143
auth_hash_hmac_sha2_384
    xform.c, 138
    xform.h, 144
auth_hash_hmac_sha2_512
    xform.c, 138
    xform.h, 144
auth_hash_key_md5
    xform.c, 138
    xform.h, 144
auth_hash_key_sha1
    xform.c, 138
    xform.h, 144
auth_hash_null
    xform.c, 139
    xform.h, 144
authctx, 15
    md5ctx, 15
    rmd160ctx, 15
    sha1ctx, 15
    sha256ctx, 15
    sha384ctx, 15
    sha512ctx, 15

blf_decrypt
    xform.c, 132
blf_encrypt
    xform.c, 132
blf_setkey
    xform.c, 132

blf_zerokey
  xform.c, 132
blocksize
  auth_hash, 13
  enc_xform, 42
BLOWFISH_BLOCK_LEN
  cryptodev.h, 85
buffer
  RMD160Context, 45

cast.c
  __FBSDID, 54
  cast_decrypt, 54
  cast_encrypt, 54
  cast_setkey, 54
  F1, 52
  F2, 52
  F3, 53
  ROL, 53
  U_INT8_Ta, 53
  U_INT8_Tb, 53
  U_INT8_Tc, 53
  U_INT8_Td, 53
cast.h
  cast_decrypt, 55
  cast_encrypt, 55
  cast_setkey, 55
CAST128_BLOCK_LEN
  cryptodev.h, 85
cast5_decrypt
  xform.c, 132
cast5_encrypt
  xform.c, 132
cast5_setkey
  xform.c, 132
cast5_zerokey
  xform.c, 133
cast_decrypt
  cast.c, 54
  cast.h, 55
cast_encrypt
  cast.c, 54
  cast.h, 55
cast_key, 17
  rounds, 17
  xkey, 17
cast_sbox1
  castsb.h, 56
cast_sbox2
  castsb.h, 56
cast_sbox3
  castsb.h, 56
cast_sbox4
  castsb.h, 56
cast_sbox5
  castsb.h, 56
cast_sbox6
  castsb.h, 56
cast_sbox7
  castsb.h, 56
cast_sbox8
  castsb.h, 57
cast_setkey
  cast.c, 54
  cast.h, 55
castsb.h
  cast_sbox1, 56
  cast_sbox2, 56
  cast_sbox3, 56
  cast_sbox4, 56
  cast_sbox5, 56
  cast_sbox6, 56
  cast_sbox7, 56
  cast_sbox8, 57
cc_alg
  cryptocap, 26
cc_arg
  cryptocap, 26
cc_flags
  cryptocap, 26
cc_freesession
  cryptocap, 26
cc_kalg
  cryptocap, 26
cc_karg
  cryptocap, 27
cc_koperations
  cryptocap, 27
cc_kprocess
  cryptocap, 27
cc_kqblocked
  cryptocap, 27
cc_max_op_len
  cryptocap, 27
cc_newsession
  cryptocap, 27
cc_process
  cryptocap, 27
cc_qblocked
  cryptocap, 27
cc_sessions
  cryptocap, 27
CIOCASYMFEAT
  cryptodev.h, 85
CIOCCRYPT
  cryptodev.h, 86
CIOCFSESSION
  cryptodev.h, 86

CIOCGSESSION
cryptodev.h, 86
CIOCKEY
cryptodev.h, 86
cipher
session_op, 46
comp_algo, 18
compress, 18
decompress, 18
minlen, 18
name, 18
type, 18
comp_algo_deflate
xform.c, 139
xform.h, 144
compress
comp_algo, 18
COP_DECRYPT
cryptodev.h, 86
COP_ENCRYPT
cryptodev.h, 86
COP_F_BATCH
cryptodev.h, 86
count
cryptotstat, 39
RMD160Context, 45
crd_alg
cryptodev.h, 86
CRD_F_COMP
cryptodev.h, 86
CRD_F_DSA_SHA_NEEDED
cryptodev.h, 87
CRD_F_ENCRYPT
cryptodev.h, 87
CRD_F_IV_EXPLICIT
cryptodev.h, 87
CRD_F_IV_PRESENT
cryptodev.h, 87
CRD_F_KEY_EXPLICIT
cryptodev.h, 87
crd_flags
cryptodesc, 29
CRD_INI
cryptodesc, 29
crd_inject
cryptodesc, 29
crd_iv
cryptodev.h, 87
crd_key
cryptodev.h, 87
crd_klen
cryptodev.h, 87
crd_len
cryptodesc, 29
crd_next
cryptodesc, 29
crd_skip
cryptodesc, 30
CRF_DH_COMPUTE_KEY
cryptodev.h, 87
CRF_DSA_SIGN
cryptodev.h, 87
CRF_DSA_VERIFY
cryptodev.h, 88
CRF_MOD_EXP
cryptodev.h, 88
CRF_MOD_EXP_CRT
cryptodev.h, 88
cri_alg
cryptoini, 31
cri_iv
cryptoini, 31
cri_key
cryptoini, 31
cri_klen
cryptoini, 31
cri_mlen
cryptoini, 31
cri_next
cryptoini, 32
CRIOGET
cryptodev.h, 88
criov.c
__FBSDID, 59
crypto_apply, 59
crypto_copyback, 59
crypto_copydata, 59
cuio_apply, 60
cuio_copyback, 60
cuio_copydata, 60
cuio_getptr, 60
CUIO_SKIP, 59
CRK_ALGORITHM_MAX
cryptodev.h, 88
CRK_ALGORITM_MIN
cryptodev.h, 88
CRK_DH_COMPUTE_KEY
cryptodev.h, 88
CRK_DSA_SIGN
cryptodev.h, 88
CRK_DSA_VERIFY
cryptodev.h, 88
crk_iparams
crypt_kop, 20
CRK_MAXPARAM
cryptodev.h, 88
CRK_MOD_EXP
cryptodev.h, 89

CRK_MOD_EXP_CRT
cryptodev.h, 89
crk_op
crypt_kop, 20
crk_oparams
crypt_kop, 20
crk_pad1
crypt_kop, 20
crk_param
crypt_kop, 20
crk_status
crypt_kop, 21
crp_buf
cryptop, 33
crp_callback
cryptop, 33
crp_desc
cryptop, 34
crp_etype
cryptop, 34
crp_flags
cryptop, 34
crp_ilen
cryptop, 34
crp_nbits
crparam, 19
crp_olen
cryptop, 34
crp_opaque
cryptop, 34
crp_p
crparam, 19
crp_sid
cryptop, 34
crp_sleep
crypto.c, 71
crp_tstamp
cryptop, 34
crparam, 19
crp_nbits, 19
crp_p, 19
crypt_kop, 20
crk_iparams, 20
crk_op, 20
crk_oparams, 20
crk_pad1, 20
crk_param, 20
crk_status, 21
crypt_op, 22
dst, 22
flags, 22
iv, 22
len, 22
mac, 22
op, 22
ses, 23
src, 23
cryptkop, 24
krp_callback, 24
krp_hid, 24
krp_iparams, 24
krp_op, 25
krp_oparams, 25
krp_param, 25
krp_status, 25
TAILQ_ENTRY, 24
crypto
crypto_if.m, 73
crypto.c
__FBSDID, 64
crp_sleep, 71
crypto_checkdriver, 64
crypto_destroy, 64
crypto_dispatch, 65
crypto_done, 65
CRYPTO_DRIVER_LOCK, 63
CRYPTO_DRIVER_UNLOCK, 63
crypto_drivers, 71
crypto_drivers_mtx, 71
crypto_drivers_num, 71
crypto_finis, 65
crypto_freereq, 66
crypto_freesession, 66
crypto_get_driverid, 66
crypto_getfeat, 66
crypto_getreq, 66
crypto_invoke, 67
crypto_kdispatch, 67
crypto_kdone, 67
crypto_kinvoke, 68
crypto_kregister, 68
crypto_mod, 71
crypto_modevent, 68
crypto_newsession, 68
crypto_proc, 69
CRYPTO_Q_LOCK, 64
CRYPTO_Q_UNLOCK, 64
crypto_register, 69
crypto_remove, 69
crypto_ret_proc, 69
CRYPTO_RETQ_EMPTY, 64
CRYPTO_RETQ_LOCK, 64
CRYPTO_RETQ_UNLOCK, 64
crypto_terminate, 70
CRYPTO_TIMING, 64
crypto_tstat, 70
crypto_unblock, 70
crypto_unregister, 70

crypto_unregister_all, 70
DECLARE_MODULE, 71
MODULE_DEPEND, 71
MODULE_VERSION, 71
TAILQ_HEAD, 71
CRYPTO_3DES_CBC
cryptodev.h, 89
CRYPTO_AES_CBC
cryptodev.h, 89
CRYPTO_ALG_FLAG_DSA_SHA
cryptodev.h, 89
CRYPTO_ALG_FLAG_RNG_ENABLE
cryptodev.h, 89
CRYPTO_ALG_FLAG_SUPPORTED
cryptodev.h, 89
CRYPTO_ALGORITHM_MAX
cryptodev.h, 89
CRYPTO_ALGORITHM_MIN
cryptodev.h, 89
crypto_apply
criov.c, 59
cryptodev.h, 97
CRYPTO_ARC4
cryptodev.h, 90
CRYPTO_ASYMQ
cryptodev.h, 90
CRYPTO_BLF_CBC
cryptodev.h, 90
CRYPTO_BUF_CONTIG
cryptodev.h, 90
CRYPTO_BUF_IOV
cryptodev.h, 90
CRYPTO_BUF_MBUF
cryptodev.h, 90
CRYPTO_CAST_CBC
cryptodev.h, 90
crypto_cdevsw
cryptodev.c, 80
crypto_checkdriver
crypto.c, 64
crypto_copyback
criov.c, 59
cryptodev.h, 97
crypto_copydata
criov.c, 59
cryptodev.h, 97
CRYPTO_DEFLATE_COMP
cryptodev.h, 90
CRYPTO_DES_CBC
cryptodev.h, 90
crypto_destroy
crypto.c, 64
crypto_dev
cryptodev.c, 81
crypto_devallowsoft
cryptodev.h, 102
crypto_dispatch
crypto.c, 65
cryptodev.h, 98
crypto_done
crypto.c, 65
cryptodev.h, 98
CRYPTO_DRIVER_LOCK
crypto.c, 63
CRYPTO_DRIVER_UNLOCK
crypto.c, 63
crypto_drivers
crypto.c, 71
CRYPTO_DRIVERS_INITIAL
cryptodev.h, 91
crypto_drivers_mtx
crypto.c, 71
crypto_drivers_num
crypto.c, 71
CRYPTO_F_BATCH
cryptodev.h, 91
CRYPTO_F_CBIFSYNC
cryptodev.h, 91
CRYPTO_F_CBIMM
cryptodev.h, 91
CRYPTO_F_DONE
cryptodev.h, 91
CRYPTO_F_IMBUF
cryptodev.h, 91
CRYPTO_F_IOV
cryptodev.h, 91
CRYPTO_F_REL
cryptodev.h, 91
crypto_finis
crypto.c, 65
crypto_freereq
crypto.c, 66
cryptodev.h, 98
crypto_freesession
crypto.c, 66
cryptodev.h, 99
crypto_get_driverid
crypto.c, 66
cryptodev.h, 99
crypto_getfeat
crypto.c, 66
cryptodev.h, 99
crypto_getreq
crypto.c, 66
cryptodev.h, 99
CRYPTO_HINT_MORE
cryptodev.h, 91
crypto_if.m

crypto, 73
crypto_invoke
crypto.c, 67
crypto_kdispatch
crypto.c, 67
cryptodev.h, 99
crypto_kdone
crypto.c, 67
cryptodev.h, 100
crypto_kinvoke
crypto.c, 68
crypto_kregister
crypto.c, 68
cryptodev.h, 100
CRYPTO_MD5
cryptodev.h, 92
CRYPTO_MD5_HMAC
cryptodev.h, 92
CRYPTO_MD5_KPDK
cryptodev.h, 92
crypto_mod
crypto.c, 71
crypto_modevent
crypto.c, 68
crypto_newsession
crypto.c, 68
cryptodev.h, 100
CRYPTO_NULL_CBC
cryptodev.h, 92
CRYPTO_NULL_HMAC
cryptodev.h, 92
CRYPTO_OP_DECRYPT
cryptodev.h, 92
CRYPTO_OP_ENCRYPT
cryptodev.h, 92
crypto_proc
crypto.c, 69
CRYPTO_Q_LOCK
crypto.c, 64
CRYPTO_Q_UNLOCK
crypto.c, 64
crypto_register
crypto.c, 69
cryptodev.h, 100
crypto_remove
crypto.c, 69
crypto_ret_proc
crypto.c, 69
CRYPTO_RETQ_EMPTY
crypto.c, 64
CRYPTO_RETQ_LOCK
crypto.c, 64
CRYPTO_RETQ_UNLOCK
crypto.c, 64
CRYPTO_RIJNDAEL128_CBC
cryptodev.h, 92
CRYPTO_RIPEMD160_HMAC
cryptodev.h, 92
CRYPTO_SESID2CAPS
cryptodev.h, 93
CRYPTO_SESID2HID
cryptodev.h, 93
CRYPTO_SESID2LID
cryptodev.h, 93
CRYPTO_SHA1
cryptodev.h, 93
CRYPTO_SHA1_HMAC
cryptodev.h, 93
CRYPTO_SHA1_KPDK
cryptodev.h, 93
CRYPTO_SHA2_256_HMAC
cryptodev.h, 93
CRYPTO_SHA2_384_HMAC
cryptodev.h, 93
CRYPTO_SHA2_512_HMAC
cryptodev.h, 94
CRYPTO_SKIPJACK_CBC
cryptodev.h, 94
CRYPTO_SW_SESSIONS
cryptodev.h, 94
CRYPTO_SYMQ
cryptodev.h, 94
crypto_terminate
crypto.c, 70
CRYPTO_TIMING
crypto.c, 64
crypto_tstat
crypto.c, 70
crypto_unblock
crypto.c, 70
cryptodev.h, 101
crypto_unregister
crypto.c, 70
cryptodev.h, 101
crypto_unregister_all
crypto.c, 70
cryptodev.h, 101
crypto_userasymcrypto
cryptodev.h, 102
crypto_usercrypto
cryptodev.h, 102
cryptocap, 26
cc_alg, 26
cc_arg, 26
cc_flags, 26
cc_freesession, 26
cc_kalg, 26
cc_karg, 27

cc_koperations, 27
cc_kprocess, 27
cc_kqblocked, 27
cc_max_op_len, 27
cc_newsession, 27
cc_process, 27
cc_qblocked, 27
cc_sessions, 27
CRYPTOCAP_F_CLEANUP
cryptodev.h, 94
CRYPTOCAP_F_SOFTWARE
cryptodev.h, 94
CRYPTOCAP_F_SYNC
cryptodev.h, 94
cryptodesc, 29
crd_flags, 29
CRD_INI, 29
crd_inject, 29
crd_len, 29
crd_next, 29
crd_skip, 30
cryptodev.c
__FBSDID, 76
crypto_cdevsw, 80
crypto_dev, 81
cryptodev_cb, 76
cryptodev_key, 76
cryptodev_mod, 81
cryptodev_modevent, 77
cryptodev_op, 77
cryptodevkey_cb, 77
cryptof_close, 77
cryptof_ioctl, 78
cryptof_kqfilter, 78
cryptof_poll, 78
cryptof_rw, 79
cryptof_stat, 79
cryptofops, 81
cryptoioctl, 79
cryptoopen, 79
cryptoread, 79
cryptowrite, 79
cseadd, 79
csecreate, 79
csedelete, 80
csefind, 80
csefree, 80
DECLARE_MODULE, 80
MODULE_DEPEND, 80
MODULE_VERSION, 80
cryptodev.h
AES_BLOCK_LEN, 85
BLOWFISH_BLOCK_LEN, 85
CAST128_BLOCK_LEN, 85
CIOCASYMFEAT, 85
CIOCCRYPT, 86
CIOCFSESSION, 86
CIOCGSESSION, 86
CIOCKEY, 86
COP_DECRYPT, 86
COP_ENCRYPT, 86
COP_F_BATCH, 86
crd_alg, 86
CRD_F_COMP, 86
CRD_F_DSA_SHA_NEEDED, 87
CRD_F_ENCRYPT, 87
CRD_F_IV_EXPLICIT, 87
CRD_F_IV_PRESENT, 87
CRD_F_KEY_EXPLICIT, 87
crd_iv, 87
crd_key, 87
crd_klen, 87
CRF_DH_COMPUTE_KEY, 87
CRF_DSA_SIGN, 87
CRF_DSA_VERIFY, 88
CRF_MOD_EXP, 88
CRF_MOD_EXP_CRT, 88
CRIOGET, 88
CRK_ALGORITHM_MAX, 88
CRK_ALGORITM_MIN, 88
CRK_DH_COMPUTE_KEY, 88
CRK_DSA_SIGN, 88
CRK_DSA_VERIFY, 88
CRK_MAXPARAM, 88
CRK_MOD_EXP, 89
CRK_MOD_EXP_CRT, 89
CRYPTO_3DES_CBC, 89
CRYPTO_AES_CBC, 89
CRYPTO_ALG_FLAG_DSA_SHA, 89
CRYPTO_ALG_FLAG_RNG_ENABLE, 89
CRYPTO_ALG_FLAG_SUPPORTED, 89
CRYPTO_ALGORITHM_MAX, 89
CRYPTO_ALGORITHM_MIN, 89
crypto_apply, 97
CRYPTO_ARC4, 90
CRYPTO_ASYMQ, 90
CRYPTO_BLF_CBC, 90
CRYPTO_BUF_CONTIG, 90
CRYPTO_BUF_IOV, 90
CRYPTO_BUF_MBUF, 90
CRYPTO_CAST_CBC, 90
crypto_copyback, 97
crypto_copydata, 97
CRYPTO_DEFLATE_COMP, 90
CRYPTO_DES_CBC, 90
crypto_devallowsoft, 102
crypto_dispatch, 98
crypto_done, 98

CRYPTO_DRIVERS_INITIAL, 91
CRYPTO_F_BATCH, 91
CRYPTO_F_CBIFSYNC, 91
CRYPTO_F_CBIMM, 91
CRYPTO_F_DONE, 91
CRYPTO_F_IMBUF, 91
CRYPTO_F_IOV, 91
CRYPTO_F_REL, 91
crypto_freereq, 98
crypto_freesession, 99
crypto_get_driverid, 99
crypto_getfeat, 99
crypto_getreq, 99
CRYPTO_HINT_MORE, 91
crypto_kdispatch, 99
crypto_kdone, 100
crypto_kregister, 100
CRYPTO_MD5, 92
CRYPTO_MD5_HMAC, 92
CRYPTO_MD5_KPDK, 92
crypto_newsession, 100
CRYPTO_NULL_CBC, 92
CRYPTO_NULL_HMAC, 92
CRYPTO_OP_DECRYPT, 92
CRYPTO_OP_ENCRYPT, 92
crypto_register, 100
CRYPTO_RIJNDAEL128_CBC, 92
CRYPTO_RIPEMD160_HMAC, 92
CRYPTO_SESID2CAPS, 93
CRYPTO_SESID2HID, 93
CRYPTO_SESID2LID, 93
CRYPTO_SHA1, 93
CRYPTO_SHA1_HMAC, 93
CRYPTO_SHA1_KPDK, 93
CRYPTO_SHA2_256_HMAC, 93
CRYPTO_SHA2_384_HMAC, 93
CRYPTO_SHA2_512_HMAC, 94
CRYPTO_SKIPJACK_CBC, 94
CRYPTO_SW_SESSIONS, 94
CRYPTO_SYMQ, 94
crypto_unblock, 101
crypto_unregister, 101
crypto_unregister_all, 101
crypto_userasymcrypto, 102
crypto_usercrypto, 102
CRYPTOCAP_F_CLEANUP, 94
CRYPTOCAP_F_SOFTWARE, 94
CRYPTOCAP_F_SYNC, 94
cuio_apply, 102
cuio_copyback, 102
cuio_copydata, 102
cuio_getptr, 102
DES3_BLOCK_LEN, 94
DES_BLOCK_LEN, 95
EALG_MAX_BLOCK_LEN, 95
HASH_MAX_LEN, 95
HMAC_IPAD_VAL, 95
HMAC_MAX_BLOCK_LEN, 95
HMAC_OPAD_VAL, 95
MALLOC_DECLARE, 102
MD5_HASH_LEN, 95
MD5_HMAC_BLOCK_LEN, 95
MD5_KPDK_HASH_LEN, 95
NULL_BLOCK_LEN, 95
NULL_HASH_LEN, 96
NULL_HMAC_BLOCK_LEN, 96
RIJNDAEL128_BLOCK_LEN, 96
RIPEMD160_HASH_LEN, 96
RIPEMD160_HMAC_BLOCK_LEN, 96
SHA1_HASH_LEN, 96
SHA1_HMAC_BLOCK_LEN, 96
SHA1_KPDK_HASH_LEN, 96
SHA2_256_HASH_LEN, 96
SHA2_256_HMAC_BLOCK_LEN, 96
SHA2_384_HASH_LEN, 96
SHA2_384_HMAC_BLOCK_LEN, 97
SHA2_512_HASH_LEN, 97
SHA2_512_HMAC_BLOCK_LEN, 97
SKIPJACK_BLOCK_LEN, 97
cryptodev_cb
  cryptodev.c, 76
cryptodev_key
  cryptodev.c, 76
cryptodev_mod
  cryptodev.c, 81
cryptodev_modevent
  cryptodev.c, 77
cryptodev_op
  cryptodev.c, 77
cryptodevkey_cb
  cryptodev.c, 77
cryptof_close
  cryptodev.c, 77
cryptof_ioctl
  cryptodev.c, 78
cryptof_kqfilter
  cryptodev.c, 78
cryptof_poll
  cryptodev.c, 78
cryptof_rw
  cryptodev.c, 79
cryptof_stat
  cryptodev.c, 79
cryptofops
  cryptodev.c, 81
cryptoini, 31
  cri_alg, 31
  cri_iv, 31

cri_key, 31
cri_klen, 31
cri_mlen, 31
cri_next, 32
cryptoioctl
cryptodev.c, 79
cryptoopen
cryptodev.c, 79
cryptop, 33
crp_buf, 33
crp_callback, 33
crp_desc, 34
crp_etype, 34
crp_flags, 34
crp_ilen, 34
crp_olen, 34
crp_opaque, 34
crp_sid, 34
crp_tstamp, 34
TAILQ_ENTRY, 33
cryptoread
cryptodev.c, 79
cryptosoft.c
__FBSDID, 105
hmac_ipad_buffer, 108
hmac_opad_buffer, 108
REGISTER, 105
swcr_authcompute, 105
swcr_authprepare, 105
swcr_compdec, 105
swcr_encdec, 106
swcr_freesession, 106
swcr_id, 109
swcr_init, 106
swcr_newsession, 107
swcr_process, 108
swcr_sesnum, 109
swcr_sessions, 109
swcr_uninit, 108
SYSUNINIT, 108
cryptosoft.h
hmac_ipad_buffer, 111
hmac_opad_buffer, 111
sw_axf, 110
sw_cxf, 110
sw_exf, 110
sw_ictx, 110
sw_klen, 110
sw_kschedule, 111
sw_mlen, 111
sw_octx, 111
sw_size, 111
cryptostats, 36
cs_blocks, 36
cs_cb, 36
cs_done, 36
cs_errs, 37
cs_finis, 37
cs_intrs, 37
cs_invoke, 37
cs_kblocks, 37
cs_kerrs, 37
cs_kops, 37
cs_ops, 37
cs_rets, 37
cryptotstat, 39
acc, 39
count, 39
max, 39
min, 39
cryptowrite
cryptodev.c, 79
cs_blocks
cryptostats, 36
cs_cb
cryptostats, 36
cs_done
cryptostats, 36
cs_errs
cryptostats, 37
cs_finis
cryptostats, 37
cs_intrs
cryptostats, 37
cs_invoke
cryptostats, 37
cs_kblocks
cryptostats, 37
cs_kerrs
cryptostats, 37
cs_kops
cryptostats, 37
cs_ops
cryptostats, 37
cs_rets
cryptostats, 37
cseadd
cryptodev.c, 79
csecreate
cryptodev.c, 79
csedelete
cryptodev.c, 80
csefind
cryptodev.c, 80
csefree
cryptodev.c, 80
csession, 40
ctxsize

auth_hash, 13
cuio_apply
criov.c, 60
cryptodev.h, 102
cuio_copyback
criov.c, 60
cryptodev.h, 102
cuio_copydata
criov.c, 60
cryptodev.h, 102
cuio_getptr
criov.c, 60
cryptodev.h, 102
CUIO_SKIP
criov.c, 59

DECLARE_MODULE
crypto.c, 71
cryptodev.c, 80
decompress
comp_algo, 18
decrypt
enc_xform, 42
deflate.c
__FBSDID, 112
deflate_global, 112
window_deflate, 113
window_inflate, 113
z_alloc, 113
z_free, 113
deflate.h
deflate_global, 115
MINCOMP, 114
z_alloc, 115
z_free, 115
Z_MEMLEVEL, 114
Z_METHOD, 114
ZBUF, 114
deflate_buf, 41
flag, 41
out, 41
size, 41
deflate_compress
xform.c, 133
deflate_decompress
xform.c, 133
deflate_global
deflate.c, 112
deflate.h, 115
des1_decrypt
xform.c, 133
des1_encrypt
xform.c, 133
des1_setkey
xform.c, 133
des1_zerokey
xform.c, 133
DES3_BLOCK_LEN
cryptodev.h, 94
des3_decrypt
xform.c, 133
des3_encrypt
xform.c, 134
des3_setkey
xform.c, 134
des3_zerokey
xform.c, 134
DES_BLOCK_LEN
cryptodev.h, 95
dst
crypt_op, 22

EALG_MAX_BLOCK_LEN
cryptodev.h, 95
enc_xform, 42
blocksize, 42
decrypt, 42
encrypt, 42
maxkey, 42
minkey, 42
name, 42
setkey, 43
type, 43
zerokey, 43
enc_xform_3des
xform.c, 139
xform.h, 144
enc_xform_arc4
xform.c, 139
xform.h, 144
enc_xform_blf
xform.c, 140
xform.h, 144
enc_xform_cast5
xform.c, 140
xform.h, 145
enc_xform_des
xform.c, 140
xform.h, 145
enc_xform_null
xform.c, 141
xform.h, 145
enc_xform_rijndael128
xform.c, 141
xform.h, 145
enc_xform_skipjack
xform.c, 141
xform.h, 145

encrypt
enc_xform, 42

F0
rmd160.c, 117
F1
cast.c, 52
rmd160.c, 117
F2
cast.c, 52
rmd160.c, 117
F3
cast.c, 53
rmd160.c, 117
F4
rmd160.c, 117
fcrypt, 44
Final
auth_hash, 14
flag
deflate_buf, 41
flags
crypt_op, 22
ftable
skipjack.c, 127

g
skipjack.c, 124
g0
skipjack.c, 125
g0_inv
skipjack.c, 125
g2
skipjack.c, 125
g2_inv
skipjack.c, 125
g4
skipjack.c, 125
g4_inv
skipjack.c, 125
g6
skipjack.c, 125
g6_inv
skipjack.c, 125
g8
skipjack.c, 126
g8_inv
skipjack.c, 126
g_inv
skipjack.c, 126

H0
rmd160.c, 117
H1
rmd160.c, 117
H2
rmd160.c, 117
H3
rmd160.c, 118
H4
rmd160.c, 118
HASH_MAX_LEN
cryptodev.h, 95
hashsize
auth_hash, 14
hmac_ipad_buffer
cryptosoft.c, 108
cryptosoft.h, 111
HMAC_IPAD_VAL
cryptodev.h, 95
HMAC_MAX_BLOCK_LEN
cryptodev.h, 95
hmac_opad_buffer
cryptosoft.c, 108
cryptosoft.h, 111
HMAC_OPAD_VAL
cryptodev.h, 95

Init
auth_hash, 14
iv
crypt_op, 22

K0
rmd160.c, 118
K1
rmd160.c, 118
K2
rmd160.c, 118
K3
rmd160.c, 118
K4
rmd160.c, 118
key
session_op, 46
keylen
session_op, 46
keysize
auth_hash, 14
KK0
rmd160.c, 118
KK1
rmd160.c, 118
KK2
rmd160.c, 119
KK3
rmd160.c, 119
KK4

rmd160.c, 119
krp_callback
cryptkop, 24
krp_hid
cryptkop, 24
krp_iparams
cryptkop, 24
krp_op
cryptkop, 25
krp_oparams
cryptkop, 25
krp_param
cryptkop, 25
krp_status
cryptkop, 25

len
crypt_op, 22

mac
crypt_op, 22
session_op, 46
mackey
session_op, 46
mackeylen
session_op, 46
MALLOC_DECLARE
cryptodev.h, 102
xform.h, 143
MALLOC_DEFINE
xform.c, 134
max
cryptotstat, 39
maxkey
enc_xform, 42
MD5_HASH_LEN
cryptodev.h, 95
MD5_HMAC_BLOCK_LEN
cryptodev.h, 95
MD5_KPDK_HASH_LEN
cryptodev.h, 95
md5ctx
authctx, 15
MD5Update_int
xform.c, 134
min
cryptotstat, 39
MINCOMP
deflate.h, 114
minkey
enc_xform, 42
minlen
comp_algo, 18
MODULE_DEPEND
crypto.c, 71
cryptodev.c, 80
MODULE_VERSION
crypto.c, 71
cryptodev.c, 80

name
auth_hash, 14
comp_algo, 18
enc_xform, 42
notreviewed.dox, 51
NULL_BLOCK_LEN
cryptodev.h, 95
null_decrypt
xform.c, 134
null_encrypt
xform.c, 134
null_final
xform.c, 134
NULL_HASH_LEN
cryptodev.h, 96
NULL_HMAC_BLOCK_LEN
cryptodev.h, 96
null_init
xform.c, 134
null_setkey
xform.c, 134
null_update
xform.c, 134
null_zerokey
xform.c, 134

op
crypt_op, 22
out
deflate_buf, 41

PADDING
rmd160.c, 121
PUT_32BIT_LE
rmd160.c, 119
PUT_64BIT_LE
rmd160.c, 119

R
rmd160.c, 119
REGISTER
cryptosoft.c, 105
RIJNDAEL128_BLOCK_LEN
cryptodev.h, 96
rijndael128_decrypt
xform.c, 135
rijndael128_encrypt
xform.c, 135

rijndael128_setkey
xform.c, 135
rijndael128_zerokey
xform.c, 135
RIPEMD160_HASH_LEN
cryptodev.h, 96
RIPEMD160_HMAC_BLOCK_LEN
cryptodev.h, 96
rmd160.c
__FBSDID, 120
F0, 117
F1, 117
F2, 117
F3, 117
F4, 117
H0, 117
H1, 117
H2, 117
H3, 118
H4, 118
K0, 118
K1, 118
K2, 118
K3, 118
K4, 118
KK0, 118
KK1, 118
KK2, 119
KK3, 119
KK4, 119
PADDING, 121
PUT_32BIT_LE, 119
PUT_64BIT_LE, 119
R, 119
RMD160Final, 120
RMD160Init, 120
RMD160Transform, 120
RMD160Update, 120
ROL, 120
X, 120
rmd160.h
RMD160_CTX, 122
RMD160Final, 122
RMD160Init, 122
RMD160Transform, 122
RMD160Update, 123
RMD160_CTX
rmd160.h, 122
RMD160Context, 45
buffer, 45
count, 45
state, 45
rmd160ctx
authctx, 15
RMD160Final
rmd160.c, 120
rmd160.h, 122
RMD160Init
rmd160.c, 120
rmd160.h, 122
RMD160Transform
rmd160.c, 120
rmd160.h, 122
RMD160Update
rmd160.c, 120
rmd160.h, 123
RMD160Update_int
xform.c, 135
ROL
cast.c, 53
rmd160.c, 120
rounds
cast_key, 17

ses
crypt_op, 23
session_op, 47
session_op, 46
cipher, 46
key, 46
keylen, 46
mac, 46
mackey, 46
mackeylen, 46
ses, 47
setkey
enc_xform, 43
SHA1_HASH_LEN
cryptodev.h, 96
SHA1_HMAC_BLOCK_LEN
cryptodev.h, 96
SHA1_KPDK_HASH_LEN
cryptodev.h, 96
sha1ctx
authctx, 15
SHA1Final_int
xform.c, 135
SHA1Init_int
xform.c, 135
SHA1Update_int
xform.c, 135
sha256ctx
authctx, 15
SHA256Update_int
xform.c, 135
SHA2_256_HASH_LEN
cryptodev.h, 96
SHA2_256_HMAC_BLOCK_LEN

cryptodev.h, 96
SHA2_384_HASH_LEN
cryptodev.h, 96
SHA2_384_HMAC_BLOCK_LEN
cryptodev.h, 97
SHA2_512_HASH_LEN
cryptodev.h, 97
SHA2_512_HMAC_BLOCK_LEN
cryptodev.h, 97
sha384ctx
authctx, 15
SHA384Update_int
xform.c, 135
sha512ctx
authctx, 15
SHA512Update_int
xform.c, 136
size
deflate_buf, 41
skipjack.c
__FBSDID, 126
ftable, 127
g, 124
g0, 125
g0_inv, 125
g2, 125
g2_inv, 125
g4, 125
g4_inv, 125
g6, 125
g6_inv, 125
g8, 126
g8_inv, 126
g_inv, 126
skipjack_backwards, 126
skipjack_forwards, 126
subkey_table_gen, 126
skipjack.h
skipjack_backwards, 128
skipjack_forwards, 128
subkey_table_gen, 128
skipjack_backwards
skipjack.c, 126
skipjack.h, 128
SKIPJACK_BLOCK_LEN
cryptodev.h, 97
skipjack_decrypt
xform.c, 136
skipjack_encrypt
xform.c, 136
skipjack_forwards
skipjack.c, 126
skipjack.h, 128
skipjack_setkey
xform.c, 136
skipjack_zerokey
xform.c, 136
src
crypt_op, 23
state
RMD160Context, 45
subkey_table_gen
skipjack.c, 126
skipjack.h, 128
sw_alg
swcr_data, 48
SW_axf
swcr_data, 48
sw_axf
cryptosoft.h, 110
SW_cxf
swcr_data, 48
sw_cxf
cryptosoft.h, 110
SW_exf
swcr_data, 49
sw_exf
cryptosoft.h, 110
SW_ictx
swcr_data, 49
sw_ictx
cryptosoft.h, 110
SW_klen
swcr_data, 49
sw_klen
cryptosoft.h, 110
SW_kschedule
swcr_data, 49
sw_kschedule
cryptosoft.h, 111
SW_mlen
swcr_data, 49
sw_mlen
cryptosoft.h, 111
sw_next
swcr_data, 49
SW_octx
swcr_data, 49
sw_octx
cryptosoft.h, 111
SW_size
swcr_data, 49
sw_size
cryptosoft.h, 111
SWCR_AUTH
swcr_data, 49
swcr_authcompute
cryptosoft.c, 105

swcr_authprepare
    cryptosoft.c, 105
SWCR_COMP
    swcr_data, 49
swcr_compdec
    cryptosoft.c, 105
swcr_data, 48
    sw_alg, 48
    SW_axf, 48
    SW_cxf, 48
    SW_exf, 49
    SW_ictx, 49
    SW_klen, 49
    SW_kschedule, 49
    SW_mlen, 49
    sw_next, 49
    SW_octx, 49
    SW_size, 49
    SWCR_AUTH, 49
    SWCR_COMP, 49
    SWCR_ENC, 49
    SWCR_UN, 49
SWCR_ENC
    swcr_data, 49
swcr_encdec
    cryptosoft.c, 106
swcr_freesession
    cryptosoft.c, 106
swcr_id
    cryptosoft.c, 109
swcr_init
    cryptosoft.c, 106
swcr_newsession
    cryptosoft.c, 107
swcr_process
    cryptosoft.c, 108
swcr_sesnum
    cryptosoft.c, 109
swcr_sessions
    cryptosoft.c, 109
SWCR_UN
    swcr_data, 49
swcr_uninit
    cryptosoft.c, 108
SYSUNINIT
    cryptosoft.c, 108

TAILQ_ENTRY
    cryptkop, 24
    cryptop, 33
TAILQ_HEAD
    crypto.c, 71
type
    auth_hash, 14
    comp_algo, 18
    enc_xform, 43

U_INT8_Ta
    cast.c, 53
U_INT8_Tb
    cast.c, 53
U_INT8_Tc
    cast.c, 53
U_INT8_Td
    cast.c, 53
Update
    auth_hash, 14

window_deflate
    deflate.c, 113
window_inflate
    deflate.c, 113

X
    rmd160.c, 120
xform.c
    __FBSDID, 132
    auth_hash_hmac_md5, 137
    auth_hash_hmac_ripemd_160, 137
    auth_hash_hmac_sha1, 137
    auth_hash_hmac_sha2_256, 137
    auth_hash_hmac_sha2_384, 138
    auth_hash_hmac_sha2_512, 138
    auth_hash_key_md5, 138
    auth_hash_key_sha1, 138
    auth_hash_null, 139
    blf_decrypt, 132
    blf_encrypt, 132
    blf_setkey, 132
    blf_zerokey, 132
    cast5_decrypt, 132
    cast5_encrypt, 132
    cast5_setkey, 132
    cast5_zerokey, 133
    comp_algo_deflate, 139
    deflate_compress, 133
    deflate_decompress, 133
    des1_decrypt, 133
    des1_encrypt, 133
    des1_setkey, 133
    des1_zerokey, 133
    des3_decrypt, 133
    des3_encrypt, 134
    des3_setkey, 134
    des3_zerokey, 134
    enc_xform_3des, 139
    enc_xform_arc4, 139
    enc_xform_blf, 140

enc_xform_cast5, 140
enc_xform_des, 140
enc_xform_null, 141
enc_xform_rijndael128, 141
enc_xform_skipjack, 141
MALLOC_DEFINE, 134
MD5Update_int, 134
null_decrypt, 134
null_encrypt, 134
null_final, 134
null_init, 134
null_setkey, 134
null_update, 134
null_zerokey, 134
rijndael128_decrypt, 135
rijndael128_encrypt, 135
rijndael128_setkey, 135
rijndael128_zerokey, 135
RMD160Update_int, 135
SHA1Final_int, 135
SHA1Init_int, 135
SHA1Update_int, 135
SHA256Update_int, 135
SHA384Update_int, 135
SHA512Update_int, 136
skipjack_decrypt, 136
skipjack_encrypt, 136
skipjack_setkey, 136
skipjack_zerokey, 136

xform.h
AH_ALEN_MAX, 143
auth_hash_hmac_md5, 143
auth_hash_hmac_ripemd_160, 143
auth_hash_hmac_sha1, 143
auth_hash_hmac_sha2_256, 143
auth_hash_hmac_sha2_384, 144
auth_hash_hmac_sha2_512, 144
auth_hash_key_md5, 144
auth_hash_key_sha1, 144
auth_hash_null, 144
comp_algo_deflate, 144
enc_xform_3des, 144
enc_xform_arc4, 144
enc_xform_blf, 144
enc_xform_cast5, 145
enc_xform_des, 145
enc_xform_null, 145
enc_xform_rijndael128, 145
enc_xform_skipjack, 145
MALLOC_DECLARE, 143

xkey
cast_key, 17

z_alloc
deflate.c, 113
deflate.h, 115

z_free
deflate.c, 113
deflate.h, 115

Z_MEMLEVEL
deflate.h, 114

Z_METHOD
deflate.h, 114

ZBUF
deflate.h, 114

zerokey
enc_xform, 43