

**NAME**

MD2, MD4, MD5, MD2\_Init, MD2\_Update, MD2\_Final, MD4\_Init, MD4\_Update, MD4\_Final, MD5\_Init, MD5\_Update, MD5\_Final - MD2, MD4, and MD5 hash functions

**SYNOPSIS**

```
#include <openssl/md2.h>

unsigned char *MD2(const unsigned char *d, unsigned long n,
unsigned char *md);

int MD2_Init(MD2_CTX *c);
int MD2_Update(MD2_CTX *c, const unsigned char *data,
unsigned long len);
int MD2_Final(unsigned char *md, MD2_CTX *c);

#include <openssl/md4.h>

unsigned char *MD4(const unsigned char *d, unsigned long n,
unsigned char *md);

int MD4_Init(MD4_CTX *c);
int MD4_Update(MD4_CTX *c, const void *data,
unsigned long len);
int MD4_Final(unsigned char *md, MD4_CTX *c);

#include <openssl/md5.h>

unsigned char *MD5(const unsigned char *d, unsigned long n,
unsigned char *md);

int MD5_Init(MD5_CTX *c);
int MD5_Update(MD5_CTX *c, const void *data,
unsigned long len);
int MD5_Final(unsigned char *md, MD5_CTX *c);
```

**DESCRIPTION**

MD2, MD4, and MD5 are cryptographic hash functions with a 128 bit output.

*MD2()*, *MD4()*, and *MD5()* compute the MD2, MD4, and MD5 message digest of the **n** bytes at **d** and place it in **md** (which must have space for MD2\_DIGEST\_LENGTH == MD4\_DIGEST\_LENGTH == MD5\_DIGEST\_LENGTH == 16 bytes of output). If **md** is NULL, the digest is placed in a static array.

The following functions may be used if the message is not completely stored in memory:

*MD2\_Init()* initializes a **MD2\_CTX** structure.

*MD2\_Update()* can be called repeatedly with chunks of the message to be hashed (**len** bytes at **data**).

*MD2\_Final()* places the message digest in **md**, which must have space for MD2\_DIGEST\_LENGTH == 16 bytes of output, and erases the **MD2\_CTX**.

*MD4\_Init()*, *MD4\_Update()*, *MD4\_Final()*, *MD5\_Init()*, *MD5\_Update()*, and *MD5\_Final()* are analogous using an **MD4\_CTX** and **MD5\_CTX** structure.

Applications should use the higher level functions *EVP\_DigestInit(3)* etc. instead of calling the hash functions directly.

**NOTE**

MD2, MD4, and MD5 are recommended only for compatibility with existing applications. In new applications, SHA-1 or RIPEMD-160 should be preferred.

**RETURN VALUES**

*MD2()*, *MD4()*, and *MD5()* return pointers to the hash value.

*MD2\_Init()*, *MD2\_Update()*, *MD2\_Final()*, *MD4\_Init()*, *MD4\_Update()*, *MD4\_Final()*, *MD5\_Init()*, *MD5\_Update()*, and *MD5\_Final()* return 1 for success, 0 otherwise.

**CONFORMING TO**

RFC 1319, RFC 1320, RFC 1321

**SEE ALSO**

[sha\(3\)](#), [ripemd\(3\)](#), [EVP\\_DigestInit\(3\)](#)

**HISTORY**

*MD2()*, *MD2\_Init()*, *MD2\_Update()*, *MD2\_Final()*, *MD5()*, *MD5\_Init()*, *MD5\_Update()* and *MD5\_Final()* are available in all versions of SSLeay and OpenSSL.

*MD4()*, *MD4\_Init()*, and *MD4\_Update()* are available in OpenSSL 0.9.6 and above.