NAME

HMAC, HMAC_CTX_init, HMAC_Init, HMAC_Init_ex, HMAC_Update, HMAC_Final, HMAC_CTX_cleanup, HMAC_cleanup - HMAC message authentication code

SYNOPSIS

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

unsigned char *HMAC(const EVP_MD *evp_md, const void *key,
int key_len, const unsigned char *d, int n,
unsigned char *md, unsigned int *md_len);

void HMAC_CTX_init(HMAC_CTX *ctx);

int HMAC_Init(HMAC_CTX *ctx, const void *key, int key_len,
const EVP_MD *md);
int HMAC_Init_ex(HMAC_CTX *ctx, const void *key, int key_len,
const EVP_MD *md, ENGINE *impl);
int HMAC_Update(HMAC_CTX *ctx, const unsigned char *data, int len);
int HMAC_Final(HMAC_CTX *ctx, unsigned char *md, unsigned int *len);

void HMAC_CTX_cleanup(HMAC_CTX *ctx);
void HMAC_cleanup(HMAC_CTX *ctx);
```

DESCRIPTION

HMAC is a MAC (message authentication code), i.e. a keyed hash function used for message authentication, which is based on a hash function.

HMAC() computes the message authentication code of the **n** bytes at **d** using the hash function **evp_md** and the key **key** which is **key_len** bytes long.

It places the result in **md** (which must have space for the output of the hash function, which is no more than **EVP_MAX_MD_SIZE** bytes). If**md** is NULL, the digest is placed in a static arra y. The size of the output is placed in **md_len**, unless it is **NULL**.

evp_md can be *EVP_sha1()*, *EVP_ripemd160()* etc.

HMAC_CTX_init() initialises a **HMAC_CTX** before first use. It must be called.

HMAC_CTX_cleanup() erases the key and other data from the **HMAC_CTX** and releases any associated resources. It must be called when an **HMAC_CTX** is no longer required.

HMAC_cleanup() is an alias for *HMAC_CTX_cleanup()* included for back compatibility with 0.9.6b, it is deprecated.

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

HMAC_Init() initializes a **HMAC_CTX** structure to use the hash function **evp_md** and the key **key** which is **key_len** bytes long. It is deprecated and only included for backward compatibility with OpenSSL 0.9.6b.

HMAC_Init_ex() initializes or reuses a **HMAC_CTX** structure to use the function **evp_md** and key **key**. Either can be NULL, in which case the existing one will be reused. *HMAC_CTX_init()* must have been called before the first use of an **HMAC_CTX** in this function. **N.B. *HMAC_Init()* had this undocumented behaviour in previous versions of OpenSSL - failure to switch to *HMAC_Init_ex()* in programs that expect it will cause them to stop working**.

HMAC_Update() can be called repeatedly with chunks of the message to be authenticated (**len** bytes at **data**).

HMAC_Final() places the message authentication code in **md**, which must have space for the hash function output.

RETURN VALUES

HMAC() returns a pointer to the message authentication code or NULL if an error occurred.

HMAC_Init_ex(), *HMAC_Update()* and *HMAC_Final()* return 1 for success or 0 if an error occurred.

HMAC_CTX_init() and *HMAC_CTX_cleanup()* do not return values.

CONFORMING TO

RFC 2104

SEE ALSO

sha(3), *evp(3)*

HISTORY

HMAC(), *HMAC_Init()*, *HMAC_Update()*, *HMAC_Final()* and *HMAC_cleanup()* are available since SSLeay 0.9.0.

HMAC_CTX_init(), *HMAC_Init_ex()* and *HMAC_CTX_cleanup()* are available since OpenSSL 0.9.7.

HMAC_Init_ex(), *HMAC_Update()* and *HMAC_Final()* did not return values in versions of OpenSSL before 1.0.0.