

NAME

ne_strhash, ne_vstrhash - string hash interface

SYNOPSIS

```
#include <ne_string.h>
```

```
char *ne_strhash(unsigned int flags, ...);
```

```
char *ne_vstrhash(unsigned int flags, va_list ap);
```

DESCRIPTION

The **ne_strhash** and **ne_vstrhash** functions can be used to create hashes. The varargs argument list must be **const char *** strings followed by a NULL terminator. The *flags* argument must select exactly one hash algorithm from the list below, which can be optionally bitwise-ORed with one of the formatting option. The hash is calculated for the concatenation of the argument list, without separators.

Hash algorithms

The following hash algorithms are available:

NE_HASH_MD5

MD5

NE_HASH_SHA256

SHA-256 (SHA-2)

NE_HASH_SHA512

SHA-512 (SHA-2)

NE_HASH_SHA256_256

SHA-512/256 (SHA-2)

Formatting options

By default, the hash is returned as a hexadecimal lower-case character string. The following formatting options are available:

NE_HASH_COLON

colon-separated hex pairs, e.g. "aa:11:22..."

NE_HASH_SPACE

space-separated hex pairs, e.g. "aa 11 22..."

RETURN VALUE

The return value is the ASCII hexadecimal representation of the hash as a malloc-allocated, NUL-terminated string, or NULL if the hash cannot be created. The string length is determined by the hash algorithm (and formatting options used). Support for hash algorithms is specific to the SSL toolkit with which neon is compiled. Some systems will further restrict hash availability at runtime, e.g. due to FIPS mode.

AUTHOR

Joe Orton

Author.

COPYRIGHT