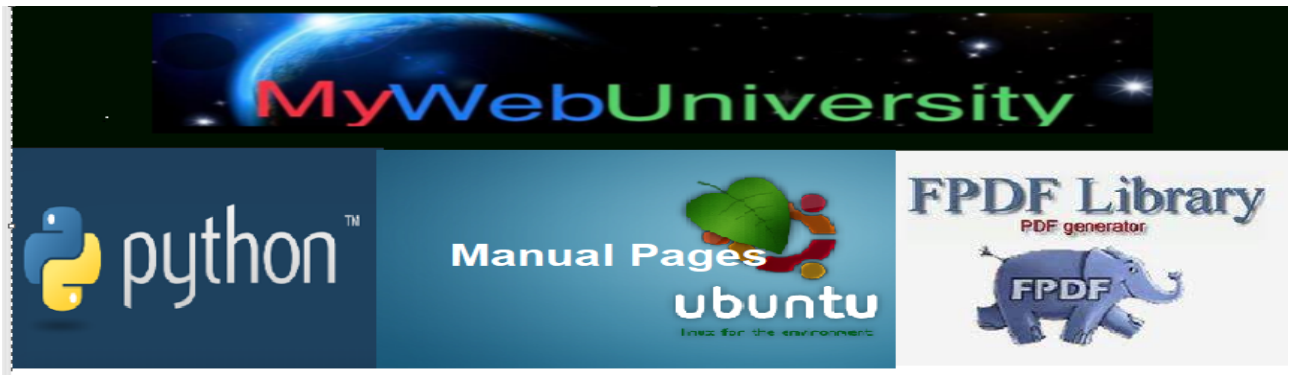

Full credit is given to the above companies including the OS that this PDF file was generated!

Linux Ubuntu 26.04 Manual Pages on command 'des_setparity.3'

$ man des_setparity.3

des_crypt(3) Library Functions Manual des_crypt(3)

NAME

des_crypt, ecb_crypt, cbc_crypt, des_setparity, DES_FAILED - fast DES encryption

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <rpc/des_crypt.h>

[[deprecated]] int ecb_crypt(unsigned int datalen;
                             char *key, char data[datalen],
                             unsigned int datalen, unsigned int mode);
[[deprecated]] int cbc_crypt(unsigned int datalen;
                             char *key, char data[datalen],
                             unsigned int datalen, unsigned int mode,
                             char *ivec);
[[deprecated]] void des_setparity(char *key);
[[deprecated]] int DES_FAILED(int status);
```

DESCRIPTION

ecb_crypt() and cbc_crypt() implement the NBS DES (Data Encryption Standard). These rou? tines are faster and more general purpose than crypt(3). They also are able to utilize DES hardware if it is available. ecb_crypt() encrypts in ECB (Electronic Code Book) mode, which encrypts blocks of data independently. cbc_crypt() encrypts in CBC (Cipher Block Chaining) mode, which chains together successive blocks. CBC mode protects against insertions, dele? tions, and substitutions of blocks. Also, regularities in the clear text will not appear in

the cipher text.

Here is how to use these routines. The first argument, key, is the 8-byte encryption key with parity. To set the key's parity, which for DES is in the low bit of each byte, use des_setparity(). The second argument, data, contains the data to be encrypted or decrypted. The third argument, datalen, is the length in bytes of data, which must be a multiple of 8. The fourth argument, mode, is formed by ORing together some things. For the encryption di? rection OR in either DES_ENCRYPT or DES_DECRYPT. For software versus hardware encryption, OR in either DES_HW or DES_SW. If DES_HW is specified, and there is no hardware, then the encryption is performed in software and the routine returns DESERR_NOHWDEVICE. For cbc_crypt(), the argument ivec is the 8-byte initialization vector for the chaining. It is updated to the next initialization vector upon return.

RETURN VALUE

DESERR_NONE

No error.

DESERR_NOHWDEVICE

Encryption succeeded, but done in software instead of the requested hardware.

DESERR_HWERROR

An error occurred in the hardware or driver.

DESERR_BADPARAM

Bad argument to routine.

Given a result status stat, the macro DES_FAILED(stat) is false only for the first two sta? tuses.

ATTRIBUTES

For an explanation of the terms used in this section, see attributes(7).

Interface	Attribute	Value
ecb_crypt(), cbc_crypt(), des_setparity()	Thread safety	MT-Safe

STANDARDS

None.

HISTORY

4.3BSD. glibc 2.1. Removed in glibc 2.28.

Because they employ the DES block cipher, which is no longer considered secure, these func?tions were removed. Applications should switch to a modern cryptography library, such as libgcrypt.

SEE ALSO

des(1), crypt(3), xcrypt(3)