

İSİM

encrypt – 64 bitlik iletileri şifreler
setkey – encrypt() tarafından kullanılan anahtarı belirler
encrypt_r – 64 bitlik iletileri şifreler (evresel)
setkey_r – encrypt_r() tarafından kullanılan anahtarı belirler (evresel)

BİLDİRİM

```
#define _XOPEN_SOURCE
#include <unistd.h>

void encrypt (char blok[64], int im);

#define _XOPEN_SOURCE
#include <stdlib.h>

void setkey (const char *anahtar);

#define _GNU_SOURCE
#include <crypt.h>

void setkey_r (const char *anahtar, struct crypt_data *veri)
void encrypt_r (char *blok, int im, struct crypt_data *veri);
```

Bunların herbiri **-lcrypt** ile ilintileme gerektirir.

AÇIKLAMA

Bu işlevler 64 bitlik iletileri şifreler ve deşifrelerler. **setkey()** işlevi **encrypt()** işlevi tarafından kullanılan anahtarı belirler. Burada kullanılan *anahtar* parametresi bir bayt dizisidir ve her bayt 1 ya da 0 sayısal değerine sahiptir. *anahtar*[*n*] dizisinin indis değeri $n=8*i-1$ olan elemanları yoksayılır, bu durumda asıl anahtar uzunluğu 56 bit olur.

setkey() işlevi kendine aktarılan tamponu *im* olarak 0 verilmişse şifreleyerek, 0 verilmişse deşifreleyerek değiştirir. *anahtar* parametresi gibi *blok* parametresi de şifrelenmiş değer bit gösteriminin vektörüdür. Sonuç aynı vektör içinde döner.

Bu iki işlev evresel değildir, yani anahtar verisi durağan bellek bölgesinde saklanır. **setkey_r()** ve **encrypt_r()** işlevleri evreseldir. ve anahtar verisini saklamak için **crypt_data** veri yapısını kullanırlar:

```
struct crypt_data {
    char keysched[16 * 8];
    char sb0[32768];
    char sb1[32768];
    char sb2[32768];
    char sb3[32768];
    char crypt_3_buf[14];
    char current_salt[2];
    long int current_saltbits;
    int direction, initialized;
};
```

DÖNÜŞ DEĞERİ

Bu işlevler herhangi bir değer döndürmezler.

HATALAR

Yukarıdaki işlevler çağrılmadan önce **errno** değişkeni sıfırlanır. Başarı durumunda değeri değişmez.

ENOSYS

İşlev kütüphanede bulunmamaktadır (Örneğin, ABD'nin ihracat sınırlamalarından dolayı).

ÖRNEK

Bu örneği glibc2.2 ile derlemek için **libcrypt** ile ilintilemeniz gerekir. Anlamalı bir çalıştırma için **anahtar[]** ve **ileti[]** dizilerini anlamalı bir bit kalıbı ile doldurmalısınız. **crypt.h** başlık dosyasının **setkey()** ve **encrypt()** işlevlerinin prototiplerini koşulsuz olarak verdiğini unutmayın.

```
#include <crypt.h>

main() {
    char anahtar[64];    /* anahtarın bit kalıbı */
    char ileti[64];      /* iletinin bit kalıbı */
    setkey(anahtar);
    encrypt(ileti, 0);    /* şifreler */
    encrypt(ileti, 1);    /* deşifreler */
}
```

NOTLAR

glibc2.2'de bu işlevler DES algoritmasını kullanır.

UYUMLULUK

encrypt() ve **setkey()** işlevleri SVID, SUSv2 ve POSIX 1003.1–2001 uyumludur. **encrypt_r()** ve **setkey_r()** işlevleri ise GNU oluşumudur.

İLGİLİ BELGELER

cbc_crypt(3), **crypt(3)**, **ecb_crypt(3)**, **fcrypt(3)**.

ÇEVİREN

Emin İslam Tatlı <eminislam@web.de>, Nisan 2004

YASAL UYARI

Bu çevirinin telif hakkı yukarıda belirtilen çevirmen(ler)e aittir. Özgün belgenin telif hakkı ve lisans bilgileri varsa ve belge içinde belirtilmemişse belge sonunda belirtilmiş olacaktır. Bu çevirinin lisansı, özgün belge için belirtilmiş bir lisans varsa ve bu lisans çevirinin de aynı lisansa sahip olmasını gerektiriyorsa onunla aynıdır, yoksa GNU GPL lisansı ve her iki durumda da ek olarak aşağıdaki koşullar geçerlidir. GNU GPL lisansı <<http://www.gnu.org/licenses/gpl.html>> adresinden edinilebilir.

BU BELGE ÜCRETSİZ OLARAK RUHSATLANDIĞI İÇİN, BELGENİN İÇERDİĞİ BİLGİLERİN VEYA KODLARIN NİTELİKLERİ İÇİN İLGİLİ KANUNLARIN İZİN VERDİĞİ ÖLÇÜDE HERHANGİ BİR GARANTİ VERİLMEMEKTEDİR. AKSİ YAZILI OLARAK BELİRTİLMEDİĞİ MÜDDETÇE TELİF HAKKI SAHİPLERİ VE/VEYA BAŞKA ŞAHISLAR BELGELERİ "OLDUĞU GİBİ", AŞIKAR VEYA ZIMNEN, SATILABİLİRLİĞİ VEYA HERHANGİ BİR AMACA UYGUNLUĞU DA DAHİL OLMAK ÜZERE HİÇBİR GARANTİ VERMEKSİZİN DAĞITMAKTADIRLAR. BELGELERİN KALİTESİ VEYA PERFORMANSI İLE İLGİLİ TÜM SORUNLAR SİZE AİTTİR. HERHANGİ BİR HATA VEYA EKSİKLİKTEN DOLAYI DOĞABİLECEK OLAN BÜTÜN SERVİS, TAMİR VEYA DÜZELTME MASRAFLARI SİZE AİTTİR.

İLGİLİ KANUNUN İCBAR ETTİĞİ DURUMLAR VEYA YAZILI ANLAŞMA HARİCİNDE HERHANGİ BİR ŞEKİLDE TELİF HAKKI SAHİBİ VEYA YUKARIDA İZİN VERİLDİĞİ ŞEKİLDE BELGEYİ DEĞİŞTİREN VEYA YENİDEN DAĞITAN HERHANGİ BİR KİŞİ, BELGENİN İÇERDİĞİ BİLGİNİN KULLANIMI VEYA KULLANILAMAMASI (VEYA VERİ KAYBI OLUŞMASI, VERİNİN YANLIŞ HALE GELMESİ, SİZİN VEYA ÜÇÜNCÜ ŞAHISLARIN ZARARA UĞRAMASI VEYA BİLGİNİN BAŞKA BİLGİLERLE UYUMSUZ OLMASI) YÜZÜNDEN OLUŞAN GENEL, ÖZEL, DOĞRUDAN YA DA DOLAYLI HERHANGİ BİR ZARARDAN, BÖYLE BİR TAZMİNAT TALEBİ TELİF HAKKI SAHİBİ VEYA İLGİLİ KİŞİYE BİLDİRİLMİŞ OLSA DAHİ, SORUMLU DEĞİLDİR.

Özgün belgedeki telif hakkı beyanı

Copyright 2000 Nicol?s Lichtmaier <nick@debian.org>

Created 2000-07-22 00:52-0300

This is free documentation; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

The GNU General Public License's references to "object code" and "executables" are to be interpreted as the output of any document formatting or typesetting system, including intermediate and printed output.

This manual is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

Modified 2002-07-23 19:21:35 CEST 2002 Walter Harms
<walter.harms@informatik.uni-oldenburg.de>

Modified 2003-04-04, aeb

Bu dosya (man3-encrypt.pdf), belgenin XML biçiminin \TeX Live ve belgeler-xsl paketlerindeki araçlar kullanılarak PDF biçimine dönüştürülmesiyle elde edilmiştir.

19 Ocak 2007