NAME

crypt − encode/decode

SYNOPSIS

crypt key

DESCRIPTION

Crypt reads from the standard input and writes on the standard output. The argument is a key that selects a particular transformation. For any given key the transformation is idempotent; that is,

```
crypt key <clear >cypher
crypt key <cypher
```

will print the clear.

The security of encrypted files depends on three factors: the fundamental method must be hard to solve; direct search of the key space must be infeasible; "sneak paths" by which keys or clear text can become visible must be minimized.

Crypt implements a one-rotor machine designed along the lines of the German Enigma, but with a 256-element rotor. Methods of attack on such machines are known, but not widely; moreover the amount of work required is likely to be large.

The transformation of a key into the internal settings of the machine is deliberately designed to be expensive, i.e. to take a substantial fraction of a second to compute. However, if keys are restricted to (say) three lower-case letters, then encrypted files can be read by expending only a substantial fraction of five minutes of machine time.

Since the key is an argument to the *crypt* command, it is potentially visible to users executing *ps*(1) or a derivative. To minimize this possibility, *crypt* takes care to destroy any record of the key immediately upon entry. No doubt the choice of keys and key security are the most vulnerable aspect of crypt.

Crypt generates files which are compatible with the **-x** option in the editor.

SEE ALSO

ed(1)