# Αλγόριθμος DES

**Δημιουργία (υπο)κλειδιών**

Ξεκινάμε με ένα κλειδί μήκους 64 bit και ένα μήνυμα το οποίο χωρίζουμε σε δέσμες 64 bit (8 bytes).

Αρχικά θέλουμε να δημιουργήσουμε τα υποκλειδιά (subkeys) που θα χρησιμοποιήσουμε στην κρυπτογράφηση. Ξεκινάμε με το αρχικό κλειδί, το οποίο περνάμε στο πρώτη μεταλλαγή (first permutation ), το οποίο αλλάζει τα bits του αντίστοιχα ( για παράδειγμα το πρώτο bit του νέου bitarray είναι το 57ο του αρχικού κλειδιού, το 2ο το 49, το 3ο το 41 κ.ο.κ.).

Στην συνέχεια, παίρνουμε το (τώρα 56 bit) “κλειδί” και το χωρίζουμε σε 2 τμήματα των 28 bit. Αυτά τα τμήματα, σε κάθε επανάληψη, μας δίνουν το κλειδί της επόμενης επανάληψης, μετά από κάποια διαδικασία.

Αρχικά, ανάλογα με τον αριθμό της επανάληψης, κάνουμε αριστερή κύλιση ενός ή δύο bit και των δύο επιμέρους κλειδιών.

Στην συνέχεια, μέσω μίας ακόμη μεταλλαγής ( second permutation ) με αντίστοιχο τρόπο, μετατρέπουμε το 56 bit κλειδί μας σε ένα κλειδί 48 bit.

Και έτσι παίρνουμε μία λίστα από κλειδιά τα οποία μπορούμε να χρησιμοποιήσουμε για να κωδικοποιήσουμε το μήνυμα.

Για την κωδικοποίηση κάθε δέσμης του αρχικού μηνύματος, ξεκινάμε με μία ακόμη μεταλλαγή ( initial permutation ) και στην συνέχεια διασπάμε την δέσμη σε δύο τμήματα 32 bit έκαστο.

Στην συνέχεια, για κάθε έναν από τους 16 “γύρους” κρυπτογράφησης, χρησιμοποιούμε μία συνάρτηση F ( [Feistel Cipher](https://en.wikipedia.org/wiki/Feistel_cipher) ), η οποία, αφού μέσω μίας συνάρτησης επέκτασης ( expansion function/table ), μετατρέπει το 32 bit δεξί τμήμα της δέσμης σε 48 bit, ώστε να είναι δυνατή η λογική πράξη αποκλειστικό ή μεταξύ της δέσμης και του αντιστοίχου δεξιού τμήμα του κλειδιού.

Στην συνέχεια, περνάμε το αποτέλεσμα από ένα “κουτί αντικατάστασης” (substitution box ή S box). Κάθε τέτοιο “κουτί” έχει σαν είσοδο 6 bits από τα 48 του προηγούμενου αποτελέσματος και έξοδο 4 bits, τα οποία και δημιουργούν το τελικό αποτέλεσμα της f, μετά από μία ακόμη μεταλλαγή P permutation.

Αυτό το αποτέλεσμα, μετά από αποκλειστικό ή με το προηγούμενο αριστερό μέρος της δέσμης ( R1 = L0 XOR f(R0, K1) ) και το αριστερό τμήμα είναι το προηγούμενο δεξί ( L1 = R0).

Συνεχίζουμε αυτήν την διαδικασία για 16 γύρους, όπου και φτάνουμε στην τελευταία μεταλλαγή του αλγορίθμου, την αντίστροφη της αρχικής μεταλλαγής ( inverse initial permutation ), όπου χρησιμοποιούμε τα 2 τμήματα του τελικού αποτελέσματος με ανάποδη σειρά, δηλαδή R16L16.

Το αποτέλεσμα αυτής της πράξης είναι και το αποτέλεσμα της κρυπτογράφησης.