Αντώνης Περής (Α.Μ.:1196)

Αρχικά, εκτός από το cs457_crypto.h και το cs457_crypto.c έχω φτιάξει αλλά 6 αρχεία (3 με βοηθητικές συναρτήσεις και 3 με τις δηλώσεις τους) και 1 main η οποία δέχεται 2 arguments από την κονσόλα: το encryption mode και το επιθυμητό testfile. Το αρχείο file_scan.c βοηθάει στην μετατροπή του αρχείου σε αλφαριθμητικό. Τα αρχεία auxiliary.c & auxiliary2.c περιέχουν βοηθητικές συναρτήσεις οι οποίες σπάνε τις κύριες συναρτήσεις της κάθε μεθόδου κρυπτογράφησης σε μικρότερα κομμάτια ώστε να είναι πιο ευανάγνωστες και μικρότερες σε μέγεθος.

Έχουν υλοποιηθεί όλες οι μέθοδοι κρυπτογράφησης όπως αναφέρονται στην εκφώνηση και στο tutorial:

1. Στην μέθοδο one time pad για την κρυπτογράφηση παράγεται ένα τυχαίο κλειδί και κάθε χαρακτήράς του plaintext γίνεται XORed με τον αντίστοιχο χαρακτήρα του κλειδιού. Η αποκρυπτογράφηση γίνεται με την αντίστροφη διαδικασία δηλαδή κάθε χαρακτήρας του ciphertext γίνεται XORed με τον αντίστοιχο χαρακτήρα του ίδιου κλειδιού, για να αποκτηθεί το αποκρηπτογραφημένο κέιμενο.
2. Στην μέθοδο ceasar’s cipher για την κρυπτογράφηση και αποκρυπτογράφηση δημιουργήθηκε ένας πίνακας όπου έχει πεζούς και κεφαλαίους λατινικούς χαρακτήρες και τα ψηφία 0-9. Η κονσόλα περιμένει input από τον χρήστη έναν αριθμό shift. Μετά για κάθε χαρακτήρα του plaintext, βρίσκεται το index για του και πραγματοποιείται μετακίνηση προς τα μπροστά ανάλογα με τον αριθμό shift. Το ολισθημένο index αν είναι μεγαλύτερο από το μέγεθος του πίνακα πραγματοποιείται wrap around. Σε κάθε περίπτωση ο κρυπτογραφημένος χαρακτήρας βρίσκεται στην θέση του μετακινημένου index. Συμμετρικά, για κάθε χαρακτήρα του ciphertext υλοποιείται και η αποκρυπτογράφηση με την διαφορά ότι εάν ο μετακινημένος προς τα πίσω index κατα shift, εάν είναι μικρότερος του 0 τότε γίνεται ανάποδο wrap around μέχρι η συνολική μετατόπιση να είναι shift.
3. Στην μέθοδο playfair μέσω της μεθόδου playfair_keymatrix ανάλογα με την passphrase δημιουργείται και ο αντίστοιχος δισδιάστατος πίνακας (5 x 5), ο οποίος αποτελεί και το κλειδί για την κρυπτογράφηση. Μετά εάν το plaintext έχει κενά χωρίζεται σε λέξεις και η κάθε λέξη μπαίνει σε μία βοηθητική συνάρτηση για να κρυπτογραφηθεί. Στην συνάρτηση αυτή χωρίζεται σε συλλαβές και εάν είναι περιττός ο αριθμός των γραμμάτων προστίθεται ένα X στο τέλος ή εάν μία συλλαβή αποτελείται από το ίδιο γράμμα 2 φορές το δεύτερο αντικαθιστάται με X. Το μορφοποιημένο plaintext μπαίνει σε μία βοηθητική συνάρτηση όπου εντοπίζονται οι συντεταγμένες βάσει το keymatrix, της κάθε συλλαβής. Εκεί για κάθε συλλαβή κρατάται η πληροφορία αν είναι στη ίδια γραμμη ή στην ίδια στηλη ή σχηματίζουν "τετράγωνο". Ανάλογα αυτής της πληροφορίας επιλέγεται και ο τρόπος κρυπτογράφησης μέσω ενός dispatch table. Οι συντεταγμένες καθώς και η παραπάνω πληροφορία κρατάται σε ένα αντικείμενο τύπου struct coordinates_of_words. Εφόσον κρυπτογραφηθούν οι συλλαβές ενώνονται για να δημιουργήσουν την κρυπτογραφημένη λέξη. Συμμέτρικά, ακολουθείται παρόμοια διαδικασία για να αποκρυπτογραφηθεί το ciphertext.
4. Στην μέθοδο κρυπτογράφισης affine χρησιμοποιείται ένας πίνακας για indexing μεγέθους 26 που έχει όλους τους κεφαλαίους λατινικούς χαρακτήρες. Ο κάθε χαρακτήρας αντιστοιχίζεται με έναν index, από τον παραπάνω πίνακα. Πιο συγκεκριμένα ο index, δίνεται σαν input στην συνάρτηση: ((((a * (x - 65)) + b) % 26) + 65). Η έξοδος της συνάρτησης αποτελεί τήν δεκαδική αναπαράσταση του κρυπτογραφημένοθ χαρακτήρα. Για την αποκρυπτογράφηση αρχικά βρίσκεται ο αντίστροφος του a(ο a_1) και μέσω της εξίσωσης (a_1 * ((x + 65 - b)) % 26) +65, υπολογίζουμε το αποκρυπτογραφημένο κείμενο.
5. Τέλος, στην μέθοδο feistel χωρίζεται το κείμενο σε κομμάτια των 8 bytes και έπειτα αυτά σπάνε σε 4 bytes για το L και 4 bytes για το R. Επίσης υλοποιήθηκε η διαδικασία round η οποία καλείται κάθε γύρο και παίρνει 2 ορίσματα: το R_i και το K (διαφορετικό σε κάθε γύρο), εκτελεί την πράξη (R_i*K) % 2^32 και το αποτέλεσμα της επιστρέφεται. Έπειτα,

ακολουθείται η διαδικασία που αναφέρεται στο φροντιστήριο για την κρυπτογράφηση και την αποκρυπτογράφηση.