Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Υπολογιστική Κρυπτογραφία

(ΣΗΜΜΥ, ΣΕΜΦΕ, ΑΛΜΑ, ΕΜΕ)

2η Σειρά Ασκήσεων

Προθεσμία παράδοσης: 26/11/2018

Άσκηση 1. Αποδείξτε ότι $(p-1)! \equiv -1 \pmod{p}$, όπου p πρώτος αριθμός.

Άσκηση 2. Υπολογίστε το $25^{-1} \bmod 77$ χωρίς αριθμομηχανή, χρησιμοποιώντας μόνο εμπειρικές παρατηρήσεις και το Κινέζικο Θεώρημα Υπολοίπων (CRT). Προτείνετε και 2ο τρόπο, χωρίς χρήση του CRT.

Άσκηση 3.
Έστω $\mathbb{Z}_p^*$ με p πρώτο και g ένας γεννήτορας, p, g γνωστά.

1. Αν d ένας ακέραιος που διαιρεί το $p-1$, βρείτε με αποδοτικό τρόπο ένα στοιχείο b του $\mathbb{Z}_p^*$ τάξης d (δηλαδή d ο μικρότερος ακέραιος με $b^d \equiv 1 \pmod{p}$)
2. Πόσα στοιχεία τάξης d υπάρχουν μέσα στο $\mathbb{Z}_p^*$;
3. Πόσους γεννήτορες έχει η κυκλική υποομάδα που παράγει ένα στοιχείο b τάξης d;
4. Πόσες κυκλικές υποομάδες τάξης d υπάρχουν στο $\mathbb{Z}_p^*$;
5. Αν μας δώσουν ένα στοιχείο h, την τάξη του d και ένα τυχαίο στοιχείο a, πώς μπορούμε να δούμε αν το a ανήκει στην υποομάδα που παράγει το h σε πολυωνυμικό χρόνο;

Άσκηση 4. Θεωρήστε την παραλλαγή του DES-X, με 2 κλειδιά k_1, k_2, όπου η κρυπτογράφηση ενός απλού κειμένου M γίνεται ως εξής :

$$Enc_{k_1,k_2}(M) = E_{k_1}(M \oplus k_2),$$

όπου E η συνάρτηση κρυπτογράφησης του DES.
Έχουμε περισσότερη ασφάλεια από τον κλασσικό DES στο παραπάνω σύστημα; Θεωρήστε ότι ο αντίπαλος έχει δυνατότητα KPA (διαθέτει αρκετά ζεύγη απλού κειμένου - κρυπτοκειμένου).

Άσκηση 5. Εξετάστε κατά πόσον το κρυπτοσύστημα AES, κατά τον τρόπο λειτουργίας CBC, διαθέτει την ιδιότητα IND-CPA. Θεωρήστε ότι το IV επιλέγεται κάθε φορά με τυχαίο ομοιόμορφο τρόπο.

Άσκηση 6. Εξετάστε τη γεννήτρια ψευδοτυχαιότητας RC4. Αποδείξτε ότι το δεύτερο byte (κλειδί) εξόδου είναι ίσο με 0 με πιθανότητα περίπου ίση με 2^{-7}. Ξεκινήστε δείχνοντας ότι αν μετά τη φάση δημιουργίας κλειδιών (KSA) ισχύει για την μετάθεση P ότι $P[2] = 0$ και $P[1] \neq 2$ τότε το δεύτερο byte εξόδου είναι ίσο 0 με πιθανότητα 1.

Άσκηση 7. Οι χρήστες $A_1, \ldots, A_n$ και B ενός δικτύου θέλουν να φτιάξουν ένα κοινό κλειδί. Όλοι θα πρέπει να γνωρίζουν αυτό το κλειδί, αλλά για κάποιον υποκλοπέα θα πρέπει να είναι αδύνατο να υπολογίσει το κλειδί. Για αυτό χρησιμοποιούν την εξής παραλλαγή του Diffie - Hellman: Έχουν για δημόσιο κλειδί έναν πρώτο αριθμό p και ένα στοιχείο $g \in \mathbb{Z}_p$ τάξης q με q πρώτο και $q|(p-1)$. Ο B διαλέγει έναν κρυφό τυχαίο αριθμό $b \in \mathbb{Z}_q^*$ και υπολογίζει το $y = g^b \pmod p$. Κάθε χρήστης A_i διαλέγει έναν κρυφό τυχαίο αριθμό $a_i \in \mathbb{Z}_q^*$ και υπολογίζει το $x_i = g^{a_i} \pmod p$. Μετά ο χρήστης A_i στέλνει το x_i στον B, που του απαντάει στέλνοντας του το $z_i = x_i^b$.

1. Δειξτε ότι κάθε χρήστης A_i έχοντας το z_i (και το a_i) μπορεί να υπολογίσει το y.
2. Δείξτε ότι το παραπάνω πρωτόκολλο είναι ασφαλές αν αποδεχτούμε την υπόθεση Diffie - Hellman. Δηλαδή αν υπάρχει αλγόριθμος που με είσοδο τις δημόσιες τιμές επιτρέπει σε έναν υποκλοπέα να υπολογισει το y, τότε μπορούμε να χρησιμοποιήσουμε τον αλγόριθμο αυτό για να καταρρίψουμε την υπόθεση Diffie - Hellman.

Άσκηση 8. (α) Παραγοντοποιήστε τον αριθμό 143 εφαρμόζοντας τη μέθοδο ρ και τη μέθοδο βάσεων παραγοντοποίησης Dixon. Τι παρατηρείτε;

(β) Υλοποιήστε σε πρόγραμμα τη μέθοδο ρ και χρησιμοποιήστε την για παραγοντοποίηση σύνθετων αριθμών της μορφής $n = pq$ με p, q πρώτους αριθμούς στο διάστημα $[10^9, 10^{10}]$. Μετρήστε το πλήθος των βημάτων σε κάθε εκτέλεση και τον συνολικό χρόνο. Τι παρατηρείτε;

Σημείωση: Για το (β) θα χρειαστεί να κατασκευάσετε πρόγραμμα που με είσοδο m να βρίσκει πρώτους αριθμούς στο διάστημα $[10^{m-1}, 10^m]$, κατά προτίμηση χρησιμοποιώντας τον έλεγχο Miller-Rabin.

Άσκηση 9. 'Εστω h συνάρτηση κατακερματισμού, η οποία συμπιέζει ακολουθίες μήκους $2n$ σε ακολουθίες μήκους n και έχει την ιδιότητα δυσκολίας εύρεσης συγκρούσεων (collision free). Θέλουμε να φτιάξουμε μία συνάρτηση κατακερματισμού που να συμπιέζει ακολουθίες μήκους $4n$ σε ακολουθίες μήκους n, η οποία να έχει επίσης την ιδιότητα δυσκολίας εύρεσης συγκρούσεων. Έχουμε τις εξής υποψήφιες:

1. $h_1(x_1||x_2||x_3||x_4) = h((x_1 \oplus x_2)||(x_3 \oplus x_4))$
2. $h_2(x_1||x_2||x_3||x_4) = h(h(x_1||x_2)||h(x_3||x_4))$
3. $h_3(x_1||x_2||x_3||x_4) = h(x_1||x_2) \oplus h(x_3||x_4)$
4. $h_4(x_1||x_2||x_3||x_4) = h(h(h(x_1||x_2)||x_3)||x_4)$

(Με "$\oplus$" συμβολίζουμε το XOR, με "||" την παράθεση και $|x_i| = n$.)

Για κάθε i εξετάστε αν η h_i έχει την ιδιότητα δυσκολίας εύρεσης συγκρούσεων ή όχι. Για να δείξετε ότι την έχει, δείξτε ότι αν μπορούσαμε να βρούμε συγκρούσεις για την h_i, τότε θα μπορούσαμε να βρούμε συγκρούσεις και για την h. Για να δείξετε το αντίθετο βρείτε μία ή περισσότερες συγκρούσεις για την h_i.

Σύντομες οδηγίες: (α) προσπαθήστε μόνοι σας, (β) συζητήστε με συμφοιτητές σας, (γ) αναζητήστε ιδέες στο διαδίκτυο, με αυτή τη σειρά και αφού αφιερώσετε αρκετό χρόνο σε κάθε στάδιο! Σε κάθε περίπτωση οι απαντήσεις πρέπει να είναι *αυστηρά ατομικές*.