Κρυπτογραφία

Ψευδοτυχαιότητα - Κρυπτοσυστήματα ροής

Άρης Παγουρτζής - Πέτρος Ποτίκας

Περιεχόμενα

Εισαγωγή

Γεννήτριες ψευδοτυχαιότητας, ψευδοτυχαίες συναρτήσεις και μεταθέσεις

Κρυπτοσυστήματα ροής (stream ciphers)

Κρυπτοσυστήματα Ροής Γραμμικής Ανάδρασης

Άλλα κρυπτοσυστήματα ροής

Εισαγωγή

Εισαγωγή

- Τυχαίοι αριθμοί αποτελούν σημαντικό στοιχείο της επιστήμης των υπολογιστών αλλά και της κρυπτογραφίας
- Αλγόριθμοι και πρωτόκολλα που τους χρησιμοποιούν:
  - Κατανομή κλειδιών, σχήματα ταυτοποίησης χρηστών
  - Ακεραιότητα μηνύματος (MAC)
  - Παραγωγή κλειδιών συνεδρίας (session keys)
  - Παραγωγή ροής από bit για συμμετρική κρυπτογράφηση (stream ciphers)

Γεννήτριες Ψευδοτυχαιότητας (Pseudorandom Generators - PRG)

- Επιτρέπουν ένα μικρό τυχαίο κλειδί (σπόρος, seed) να δώσει ένα μεγάλο “ψευδοτυχαίο” string, αρκετά τυχαίο για έναν πολυωνυμικά φραγμένο αντίπαλο.
- Το ψευδοτυχαίο string μπορεί να χρησιμοποιηθεί σαν κλειδί για το one-time pad (πράξη XOR).
- Παρεμφερής χρήση: σε κρυπτοσυστήματα ροής.
- Η ύπαρξη ψευδοτυχαίων γεννητριών σχετίζεται με την ύπαρξη μονόδρομων συναρτήσεων (one-way functions).
- RC4 (Rivest ’87): μια σημαντική γεννήτρια / κρυπτοσύστημα ροής.

Γεννήτριες ψευδοτυχαιότητας, ψευδοτυχαίες συναρτήσεις και μεταθέσεις

Γεννήτριες Ψευδοτυχαιότητας

- Ιδέα: κάτι που να “μοιάζει” με τυχαίο, χωρίς να είναι πραγματικά
- Τυχαία string να μη διακρίνονται από string που παράγονται από τη γεννήτρια ψευδοτυχαιότητας
- Εφαρμογές ψευδοτυχαιότητας: παίγνια, δειγματοληψία, πιθανοτικοί αλγόριθμοι
- Κρυπτογραφική χρήση: παραγωγή κλειδιών σε σχήματα συμμετρικής/ασύμμετρης κρυπτογράφησης, κρυπτοσυστήματα ροής

Γεννήτριες Ψευδοτυχαιότητας (PRG)

Ποιο είναι πιο “τυχαίο”;

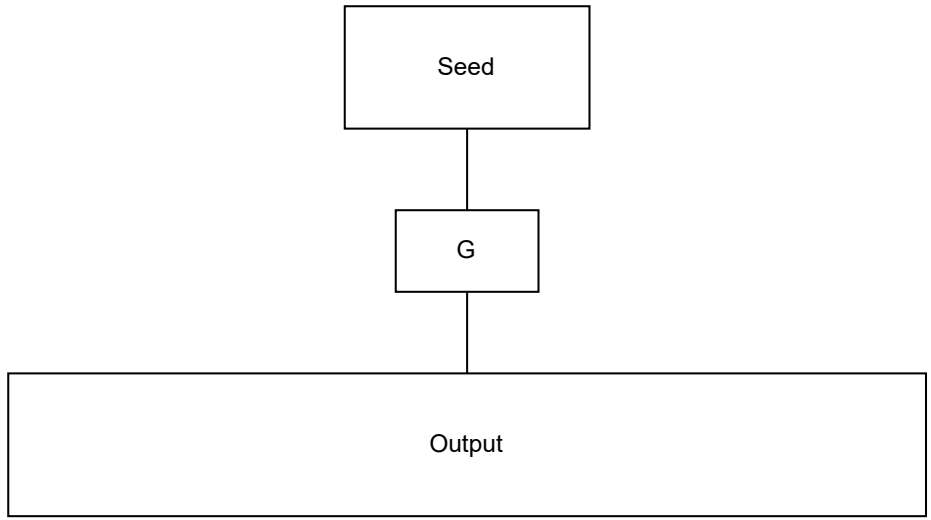
00101010100101010110

01010101010101010101

- Ορισμός ψευδοτυχαιότητας μέσω στατιστικών τεστ: Μια κατανομή πιθανότητας *Dist* πάνω σε n-bit strings είναι ψευδοτυχαία αν ικανοποιεί κάποια τεστ (NIST SP 800-22)
  1. $\Pr_{x \leftarrow Dist}[\text{πρώτο bit του } x = 1] \simeq 1/2$
  2. $\Pr_{x \leftarrow Dist}[\text{parity του } x = 1] \simeq 1/2$
  3. $\Pr_{x \leftarrow Dist}[\#1 = \#0 \text{ in } x] \simeq 1/2$
  4. ...
- Όμως δε γνωρίζουμε τα τεστ που μπορεί να έχει ο αντίπαλος
- Κρυπτογραφικά, η κατανομή *Dist* είναι ψευδοτυχαία, αν περνάει *όλα* τα αποδοτικά στατιστικά τεστ

Γεννήτριες Ψευδοτυχαιότητας (PRG)

Μια γεννήτρια ψευδοτυχαιότητας (PRG) είναι ένας αποδοτικός, ντετερμινιστικός αλγόριθμος που επεκτείνει ένα *μικρό, ομοιόμορφα τυχαία επιλεγμένο σπόρο (seed)* σε μια μεγαλύτερη, ψευδοτυχαία έξοδο.



Γεννήτριες Ψευδοτυχαιότητας (PRG)

- Από λίγα πραγματικά τυχαία bits, παράγονται πολλά περισσότερα bits που “φαίνονται” τυχαία
- Παραγωγή “πραγματικά” τυχαίων bits είναι δύσκολη και χρονοβόρα (ίσως και ανέφικτη)
- Επιλογή σπόρου: μεγάλο μήκος, μη προβλεψιμότητα
- Όπως η σημασιολογική (υπολογιστική) ασφάλεια είναι η υπολογιστική χαλάρωση της τέλειας μυστικότητας έτσι και η ψευδοτυχαιότητα είναι η υπολογιστική χαλάρωση της πραγματικής τυχαιότητας.

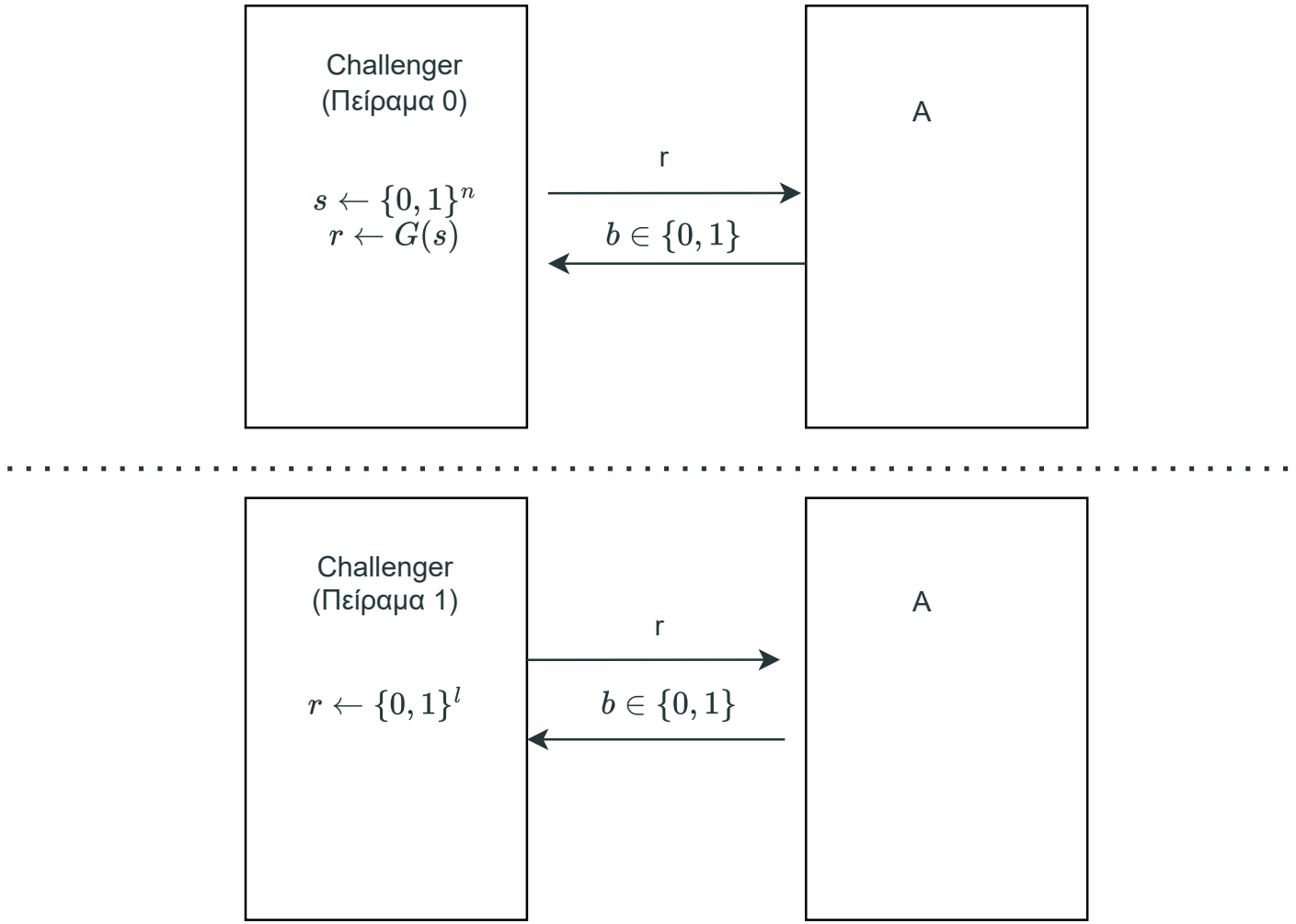
Τυπικός ορισμός ψευδοτυχαίας κατανομής

Έστω συνάρτηση $G : \{0,1\}^n \mapsto \{0,1\}^l$.
Ορίζουμε Dist την κατανομή πάνω σε l-bit strings που προκύπτει επιλέγοντας ομοιόμορφα τυχαία ένα $s \in \{0,1\}^n$ και παίρνοντας το $G(s)$.
Η G είναι ψευδοτυχαία ανν η Dist είναι ψευδοτυχαία.

Πείραμα

- Θεωρούμε πως έχουμε έναν πολυωνυμικά φραγμένο αντίπαλο, ο οποίος λαμβάνει strings μήκους l.
- Θέλουμε ο αντίπαλος να μην καταλαβαίνει αν παίρνουμε δείγμα από την κατανομή Dist ή αν παίρνουμε ομοιόμορφα τυχαία l-bit string
- Θέλουμε ο αντίπαλος να μην καταλαβαίνει αν αυτά προήλθαν από την $G(s)$ (με ομοιόμορφα τυχαία επιλεγμένο s) ή αν αυτά προήλθαν ομοιόμορφα τυχαία από το $\{0,1\}^l$ (δηλ. είναι πραγματικά τυχαία string μήκους l)

Πείραμα Ψευδοτυχαιότητας



Γεννήτριες Ψευδοτυχαιότητας

Ορισμός

Έστω l πολυώνυμο, G ένας ντετερμινιστικός αλγόριθμος πολυωνυμικού χρόνου, τ.ώ. για κάθε n και είσοδο $s \in \{0,1\}^n$ το αποτέλεσμα $G(s)$ είναι μήκους $l(n)$. Ο G είναι *γεννήτρια ψευδοτυχαιότητας (PRG)* αν:

1. Για κάθε n, $l(n) > n$
2. Για κάθε πιθανοτικό πολυωνυμικού χρόνου αλγόριθμο (PPT) D, υπάρχει μια αμελητέα[1] συνάρτηση $negl$, ώστε

$$|Pr_{s \leftarrow \{0,1\}^n}[D(G(s)) = 1] - Pr_{r \leftarrow \{0,1\}^{l(n)}}[D(r) = 1]| \leq negl(n)$$

όπου η πρώτη πιθανότητα είναι πάνω στην ομοιόμορφα τυχαία επιλογή $s \in \{0,1\}^n$ και την τυχαιότητα του D, η δεύτερη πάνω στην ομοιόμορφα τυχαία επιλογή $r \in \{0,1\}^{l(n)}$ και την τυχαιότητα του D.

[1] αμελητέα συνάρτηση f: για κάθε πολυώνυμο p, υπάρχει μια σταθερά n_0, τ.ώ. για κάθε $n > n_0$ ισχύει $f(n) < 1/p(n)$

Γεννήτριες Ψευδοτυχαιότητας

Παράδειγμα

- Δίνεται η γεννήτρια
  $G(s) = s|| \oplus_{i=1}^{n} s_i$
- Είναι PRG;

Γεννήτριες Ψευδοτυχαιότητας

Παρατηρήσεις:

- Ο αλγόριθμος είναι ντετερμινιστικός και αποδοτικός (πολυωνυμικός)
- Είναι τυχαία η κατανομή;
- αν $l(n) = n + 1$, τότε στην ομοιόμορφη κατανομή στο $\{0,1\}^{n+1}$ κάθε συμβολοσειρά έχει ακριβώς $1/2^{n+1}$ πιθανότητα να επιλεγεί
- αν $|dom(G)| = 2^n$, τότε η πιθανότητα μια συμβολοσειρά μήκους $n + 1$ να εμφανιστεί στην έξοδο της G είναι τουλάχιστον $1/2^n$ για τις μισές το πολύ συμβολοσειρές και 0 για τις υπόλοιπες
- Αν ο διαχωριστής είναι εκθετικού χρόνου, τότε με εξαντλητική αναζήτηση μπορεί να ξεχωρίσει την κατανομή D από την ομοιόμορφη
- Ο σπόρος πρέπει να μείνει μυστικός και αρκετά μεγάλος, ώστε να μη γίνεται επίθεση με εξαντλητική αναζήτηση
- Έστω διαχωριστής που για δοσμένο $r = r_1 r_2 \dots r_{n+1}$ ελέγχει αν $r_{i+1} = \oplus_{i=1}^{n} r_i$. Τι πλεονέκτημα έχει; (άσκηση)

Γεννήτριες Ψευδοτυχαιότητας

- Υπάρχουν γεννήτριες ψευδοτυχαιότητας; Άγνωστο, χωρίς κάποια υπόθεση.
- Μπορούν να κατασκευαστούν με την υπόθεση ότι υπάρχουν μονόδρομες συναρτήσεις (one-way functions).
- Υπάρχουν, με την υπόθεση ότι το πρόβλημα της παραγοντοποίησης μεγάλων αριθμών είναι δύσκολο.
- Υποψήφιες: stream ciphers, block ciphers (OFB, CFB, CTR mode)

Γεννήτριες Ψευδοτυχαιότητας και μη προβλέψιμότητα

- Στο σημείο αυτό θα θεωρήσουμε την περίπτωση που η γεννήτρια παράγει μια ακολουθία από τυχαία bits.
- Ισχύει: G γεννήτρια ψευδοτυχαιότητας ανν G μη προβλέψιμη

Ορισμός

(Προβλέψιμη) Υπάρχει πολυωνυμικός αλγόριθμος A τέτοιος ώστε:

$$Pr[A(G(K)_{1..i}) = G(K)_{i+1}] > \frac{1}{2} + \epsilon$$

για μη αμελητέο ϵ — probability amplification!

Επιπλέον, θα πρέπει να έχουμε και προς τα πίσω μη προβλεψιμότητα: οι τιμές που έχουν εμφανιστεί δεν αποκαλύπτουν το σπόρο.

One-time pad: OTP

- Ένα καλό σύστημα κρυπτογράφησης:
  $|M| = |K| = |C| = \{0,1\}^n$,

$$Enc_k(m) = k \oplus m, Dec_k(c) = k \oplus c$$

- Το OTP έχει τέλεια μυστικότητα, αλλά πρέπει
  $|key| = |plaintext|$
- Μη ρεαλιστικό!

PRG $\rightarrow$ Secure cipher

- Ιδέα: από ένα μικρό, πραγματικά "τυχαίο" σπόρο (seed) φτιάχνω ένα μεγάλο, "ψευδοτυχαίο" κλειδί, έτσι κρυπτογραφώ μεγάλου μεγέθους δεδομένα:

$$c = Enc_k(m) = G(k) \oplus m$$

$$m = Dec_k(c) = G(k) \oplus c$$

- G ντετερμινιστική συνάρτηση πολυωνυμικού χρόνου με $|G(k)| = p(|k|)$
- Απόδειξη ασφάλειας: υπόθεση πως η G είναι ψευδοτυχαία (αναγωγή)

Απόδειξη ασφάλειας

Θεώρημα

Αν G είναι μια γεννήτρια ψευδοτυχαιότητας, τότε το παραπάνω σχήμα κρυπτογράφησης έχει μη διακρίσιμες κρυπτογραφήσεις στο μοντέλο παθητικού αντιπάλου (IND-EAV).

Απόδειξη.

(Ιδέα) Με αναγωγή: Υποθέτουμε ότι έχουμε αντίπαλο $\mathcal{A}$ ο οποίος διακρίνει τις κρυπτογραφήσεις. Χρησιμοποιώντας τον $\mathcal{A}$ ως μαντείο μπορούμε να διακρίνουμε την έξοδο της G από μία πραγματικά τυχαία ως εξής:
Με είσοδο $w \in \{0,1\}^{l(n)}$ (που προήλθε από την G ή είναι τυχαίο) ζητάμε από τον $\mathcal{A}$ δύο plaintext m_0, m_1, επιλέγουμε $b \in \{0,1\}$ και του στέλνουμε $c = m_b \oplus w$. Αν ο $\mathcal{A}$ επιστρέψει $b = b'$ λέμε ότι το w προήλθε από την G. Αν ο $\mathcal{A}$ έχει πλεονέκτημα, τότε έχουμε και εμείς. Άτοπο. □

Pseudorandom Functions - PRF

- Συνάρτηση που φαίνεται ίδια με μια τυχαία συνάρτηση
- Τυχαία συνάρτηση: $Func_n$ = όλες οι συναρτήσεις από το $\{0,1\}^n$ στο $\{0,1\}^n$
- Πόσες; Μπορούμε να αναπαραστήσουμε μια συνάρτηση στο $Func_n$ με $n2^n$ bits
- Άρα, $|Func_n| = 2^{n2^n}$
- Τυχαία συνάρτηση: διάλεξε ομοιόμορφα μια $f \in Func_n$
- Ισοδύναμα: σε κάθε θέση του πίνακα τιμών διάλεξε ομοιόμορφα ένα string από το $\{0,1\}^n$

Pseudorandom Functions - PRF

- Δεν έχει νόημα να μιλάμε για σταθερή συνάρτηση, αλλά θέλουμε κάποια κατανομή
- Αν έχουμε μια $F\colon \{0,1\}^* \times \{0,1\}^* \mapsto \{0,1\}^*$, τότε αν κρατήσουμε σταθερή την πρώτη παράμετρο έχουμε συναρτήσεις $F_k(x) = F(k,x)$, όπου k κλειδί (επιλέγεται ομοιόμορφα)
- Επιλέγοντας το κλειδί $k \leftarrow \{0,1\}^n$ επιλέγεται μια $F_k\colon \{0,1\}^n \to \{0,1\}^n$
- Άρα η F με κλειδί, ορίζει μια κατανομή στις συναρτήσεις της $Func_n$

Τυπικός ορισμός PRF

- Η αναπαράσταση με $n2^n$ bits είναι αδύνατο να ελεχθεί από έναν πολυωνυμικό διαχωριστή
- Έχουμε ένα μαντείο O που είτε είναι ίσο με F_k (για ομοιόμορφο k) ή με f (για ομοιόμορφη f)
- Μπορούμε να ρωτήσουμε για όποιο x θέλουμε, αλλά ίδια απάντηση για το ίδιο x.
- Μόνο πολυωνυμικά πολλές ερωτήσεις γίνονται στο μαντείο. Οι ερωτήσεις προσαρμόζονται.

Ψευδοτυχαία συνάρτηση - Ορισμός

Ορισμός

Έστω συνάρτηση $F : \{0,1\}^* \times \{0,1\}^* \mapsto \{0,1\}^*$ αποδοτικά υπολογίσιμη, με κλειδί. Η F είναι *ψευδοτυχαία συνάρτηση* αν για κάθε πιθανοτικό πολυωνυμικού χρόνου διαχωριστή D υπάρχει αμελητέα συνάρτηση $negl$ ώστε:

$$|Pr_{k\leftarrow\{0,1\}^n}[D^{F_k()}(1^n) = 1] - Pr_{f\leftarrow Func_n}[D^{f()}(1^n) = 1]| \leq negl(n)$$

όπου η πρώτη πιθανότητα είναι πάνω στην τυχαία επιλογή του κλειδιού $k \in \{0,1\}^n$ και την τυχαιότητα του D, ενώ η δεύτερη ως προς την τυχαία επιλογή της $f \in Func_n$ και την τυχαιότητα του D

Σημείωση: Αν δοθεί το κλειδί, παύει να είναι PRF.

Παράδειγμα

$F(k, x) = k \oplus x$. Είναι ψευδοτυχαία συνάρτηση;

Ψευδοτυχαία μετάθεση (Pseudorandom permutation)

- Υπάρχει και η έννοια της ψευδοτυχαίας μετάθεσης, δηλ. συνάρτηση που είναι 1-1 και επί (άρα έχει και αντίστροφη)
- Ο υπολογισμός της αντίστροφης πρέπει να γίνεται αποδοτικά.
- Όμως έχουμε oracle και για την αντίστροφη, οπότε ο ορισμός της ασφάλειας πρέπει να αλλάξει (strong pseudorandom permutation)

Ορισμός

Έστω 1-1 και επί συνάρτηση $F : \{0,1\}^* \times \{0,1\}^* \mapsto \{0,1\}^*$, αποδοτικά υπολογίσιμη, με κλειδί. Η F είναι *ισχυρά ψευδοτυχαία μετάθεση* αν για κάθε πιθανοτικό πολυωνυμικού χρόνου διαχωριστή D υπάρχει αμελητέα συνάρτηση *negl* ώστε:

$$|Pr_{k\leftarrow\{0,1\}^n}[D^{F_k(),F_k^{-1}()}(1^n) = 1] - Pr_{f\leftarrow Func_n}[D^{f(),f^{-1}()}(1^n) = 1]| \leq negl(n)$$

PRF $\leftrightarrow$ PRG

Από μια ψευδοτυχαία συνάρτηση μπορούμε να πάρουμε μια ψευδοτυχαία γεννήτρια: $G(k) = F_k(0)||F_k(1)||F_k(2)...$

Αλλά και αντίστροφα, από μια PRG μπορούμε να πάρουμε μια PRF:

Έστω PRG G με παράγοντα επέκτασης $n2^{t(n)}$, τότε ορίζεται μια συνάρτηση $F : \{0,1\}^n \times \{0,1\}^{t(n)} \mapsto \{0,1\}^n$
Για να υπολογίσουμε το $F_k(i)$, υπολογίζουμε το $G(k)$ και ερμηνεύουμε το αποτέλεσμα σαν look-up table με $2^{t(n)}$ γραμμές, όπου κάθε γραμμή περιέχει ένα n-bit string. Δίνουμε ως έξοδο την i-οστή γραμμή.

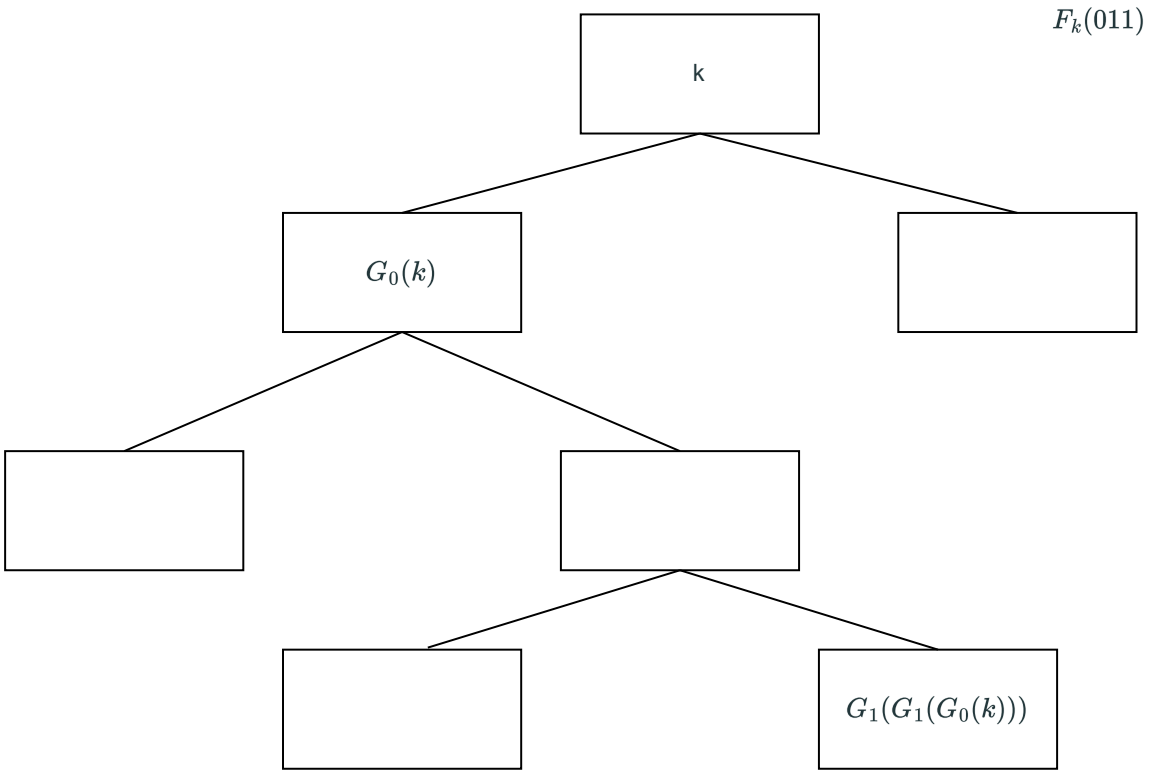
PRG $\rightarrow$ PRF (καλύτερα)

Πιο αποδοτικός τρόπος:

Έστω $G : \{0,1\}^n \mapsto \{0,1\}^{2n}$ PRG.

$G(k) = G_0(k)||G_1(k)$, με $G_0(k), G_1(k)$ τα δύο μισά της εξόδου.

Για $k \in \{0,1\}^n$, ορίζουμε $F_k(x_1x_2...x_n) = G_{x_n}(...G_{x_2}(G_{x_1}(k)))$



Δημιουργία πραγματικής τυχαιότητας

- υλικό, φυσικά φαινόμενα π.χ. θερμικός ή ηλεκτρικός θόρυβος
- λογισμικό π.χ. πάτημα πλήκτρων πληκτρολογίου, κίνηση του ποντικιού

Γεννήτριες τυχαίων αριθμών γενικού σκοπού είναι μη κατάλληλες για την κρυπτογραφία π.χ. rand() της C.

Intel, random.org ...

‘Αποδεδειγμένα’ ασφαλείς γεννήτριες ψευδοτυχαίων

- RSA-based (Micali-Schnorr), BBS.
- Βασίζονται σε (γενικά παραδεκτές) αριθμοθεωρητικές μονόδρομες συναρτήσεις: ύψωση σε δύναμη modulo n, τετραγωνισμός modulo n.
- Λειτουργία: διαδοχικές εφαρμογές της συνάρτησης, έξοδος κάθε φορά το λιγότερο σημαντικό bit του αριθμού (ή κάποια από τα λιγότερο σημαντικά bit).
- Είναι ασφαλείς κάτω από την υπόθεση δυσκολίας αντιστροφής της αντίστοιχης συνάρτησης.
- Απαιτούν μεγαλύτερη υπολογιστική προσπάθεια.

Γεννήτρια BBS [Blum-Blum-Shub (1986)]

Αλγόριθμος

- Βρες δύο μεγάλους πρώτους p, q (μυστικά), με $p \equiv q \equiv 3 \pmod 4$, και θέσε $n = pq$ (n: Blum integer).
- Επίλεξε τυχαία ένα s_0 σχετικά πρώτο με το n.
- Δώσε έξοδο (i-οστό bit):

$$z_i = (s_0^{2^i} \bmod n) \bmod 2$$

για $1 \leq i \leq \infty$

Παρατήρηση: σχετικά αργή, αλλά ασφαλής υπό την υπόθεση ότι ο έλεγχος τετραγωνικών υπολοίπων $\pmod n$ είναι δύσκολος αν δεν είναι γνωστή η παραγοντοποίηση του n.

Παράδειγμα BBS

Έστω $n = 192649 = 383 * 503$ και $s_0 = 101355^2 \mod n = 20749$.

Τα πρώτα 5 bits που παράγονται από τον BBS είναι

11001

και προκύπτουν:

i	s_i	z_i
0	20749	
1	143135	1
2	177671	1
3	97048	0
4	89992	0
5	174051	1

Ιδιότητες BBS

- 'Ζει' στην ομάδα $QR(n)$.
- Η πράξη τετραγωνισμού $x \mapsto x^2 \pmod{n}$ είναι μετάθεση στο $QR(n)$ όταν n είναι Blum integer.
- Εύρεση $s_0^{2^{i-1}}$ από $s_0^{2^i}$ αν δεν γνωρίζουμε p, q: ισοδύναμη με παραγοντοποίηση.
- Εύρεση $s_0^{2^{i-1}}$ από $s_0^{2^i}$ αν *γνωρίζουμε* p, q: εφικτή, χρήση σε *πιθανοτική κρυπτογράφηση δημοσίου κλειδιού Goldwasser-Micali*.
- Επιτάχυνση: μπορεί να χρησιμοποιηθούν έως και $(\log \log n)$ bits σε κάθε βήμα.

Η γεννήτρια ψευδοτυχαίων RC4

- Rivest (1987)
- Ιδιωτικό της εταιρίας RSA Data Security, Inc (κλειστό)
- Διέρρευσε το 1994
- Χρήση σε πολύ διαδεδομένα πρωτόκολλα: WEP/WPA, SSL/TLS

Η γεννήτρια ψευδοτυχαίων RC4

- Συστατικά: 2 arrays of bytes:
  - Μετάθεση $P[0..255]$. Αρχικοποίηση:
    for all $i \in \{0..255\}$ **do** : $P[i] = i$
  - Κλειδί $K[0..keylen - 1]$, $keylen \leq 256$ – συνήθως $keylen \in [5..8]$.
    Επιλέγεται από χρήστη.
- Δημιουργία σειράς κλειδιών (key-scheduling algorithm – KSA).

  Η αρχική (ταυτοτική) μετάθεση P μετατρέπεται μέσω μιας σειράς ανταλλαγών (swap) σε μια φαινομενικά τυχαία μετάθεση.

  Το "ανακάτεμα" επηρεάζεται από το αρχικό κλειδί K και τις μεταβολές της P.
- Παραγωγή ψευδοτυχαίων bytes (pseudorandom generation algorithm – PRGA)

  Επαναληπτικός βρόχος: σε κάθε επανάληψη επιλέγεται κάποιο byte της P ως κλειδί εξόδου με τρόπο που καθορίζεται από τα τρέχοντα περιεχόμενα της P.

  Οι επαναλήψεις συνεχίζονται για όσο χρειάζεται (δηλ. μέχρι να τελειώσει το stream). Σε κάθε επανάληψη γίνεται και ένα νέο swap στοιχείων της P.

Η γεννήτρια ψευδοτυχαίων RC4

Περιγραφή KSA και PRGA

- Δημιουργία σειράς κλειδιών (KSA)

  $j = 0$

  for $i = 0$ **to** 255 **do** :

  $j = (j + P[i] + K[i \bmod keylen]) \bmod 256$

  $\mathrm{swap}(P[i], P[j])$

- Παραγωγή ψευδοτυχαίων bytes (PRGA)

  $i = 0;\ j = 0$

  while next key needed :

  $i = (i + 1) \bmod 256\ ;\ j = (j + P[i]) \bmod 256$

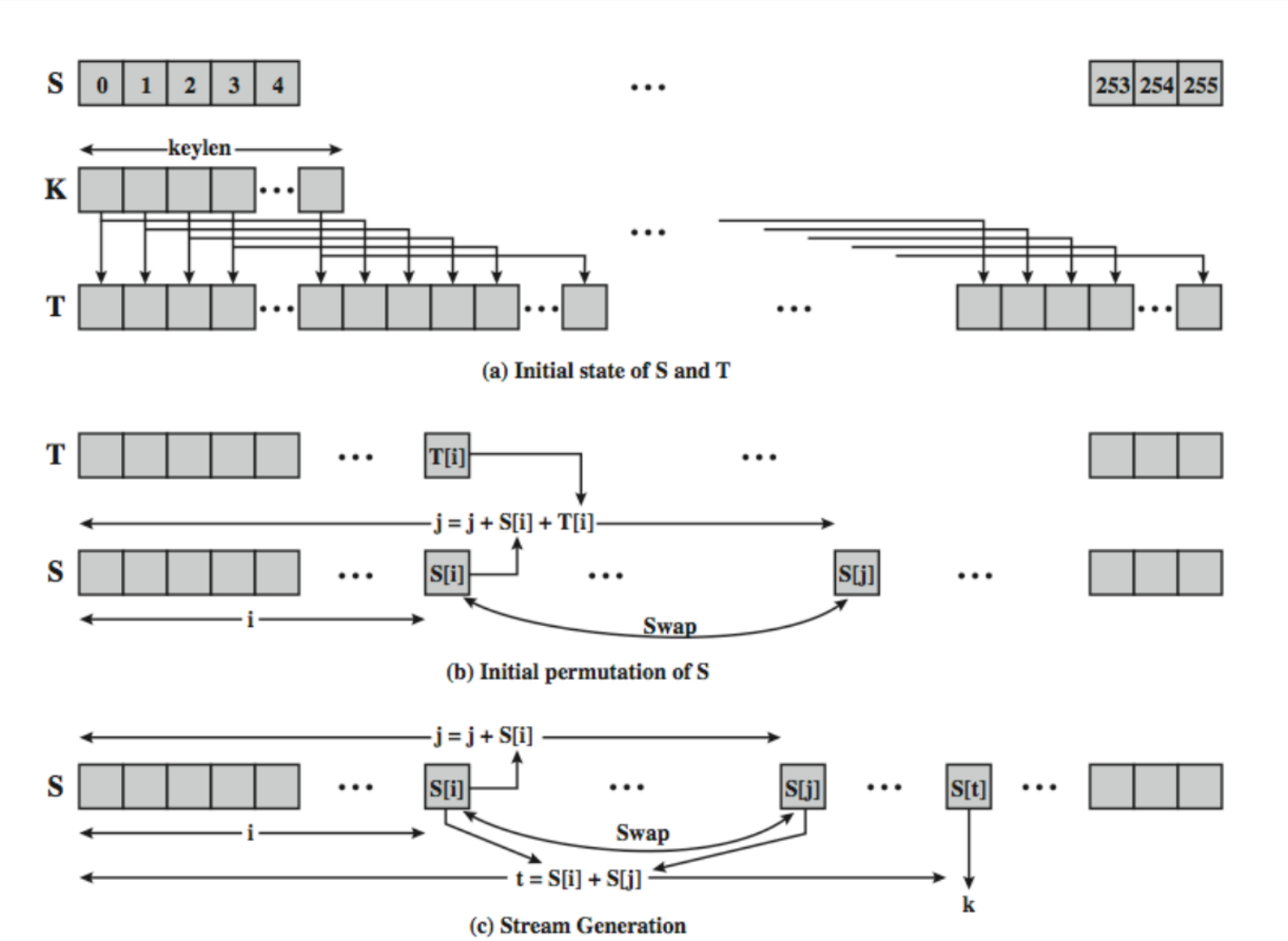
  $\mathrm{swap}(P[i], P[j])$

  $K_o = P[(P[i] + P[j]) \bmod 256]$

  output K_o

Κάθε κλειδί εξόδου K_o χρησιμοποιείται για την κρυπτογράφηση ενός byte αρχικού κειμένου.

RC4 σχηματικά



Σχήμα 4: RC4

Η γεννήτρια ψευδοτυχαίων RC4

Παρατηρήσεις

- Με ίδιο αρχικό κλειδί K προκύπτει η ίδια σειρά κλειδιών εξόδου.
- Απλή και γρήγορη στην υλοποίηση με software (σε αντίθεση με άλλα stream cipher, π.χ. αυτά που βασίζονται σε LFSRs).
- Η ασφάλεια της γεννήτριας RC4 έχει αμφισβητηθεί έντονα. Κάποιοι τρόποι χρήσης ιδιαίτερα ανασφαλείς (π.χ. WEP) – επίθεση Fluhrer, Mantin, Shamir (2001).
- Ουσιαστικό πρόβλημα η παραλλαγή του RC4 με χρήση IV, όπου μπορεί να αποκαλυφθεί το πραγματικό κλειδί (WEP)
- Μη ασφαλής!
- Άμυνα: απόρριψη αρχικού τμήματος κλειδοροής (RC4-drop[n]), ενδεικτικά: $n = 768$ bytes, συστήνεται ακόμη και $n = 3072$.

Κρυπτοσυστήματα ροής (stream ciphers)

Κρυπτοσυστήματα ροής (stream ciphers)

Παραγωγή ακολουθίας κλειδιών με βάση κάποιο αρχικό κλειδί, και (πιθανά) το plaintext.

Ορισμός

- Plaintext: $x_0, x_1, \ldots, x_{n-1}$
- Ciphertext: $y_0, y_1, \ldots, y_{n-1}$
- Αρχικό κλειδί: k
- Βοηθητικές συναρτήσεις: $f_i, 0 \leq i < m$
- Key stream: $z_i = f_{i \bmod m}(k, x_0, \ldots, x_{i-1}, z_0, \ldots, z_{i-1})$
- Κρυπτογράφηση: $y_i = enc_{z_i}(x_i)$
- Αποκρυπτογράφηση: $x_i = dec_{z_i}(y_i)$

Π.χ. για δυαδικές ακολουθίες:

$$enc_z(x) = x \oplus z = x + z \bmod 2$$
$$dec_z(y) = y \oplus z = y + z \bmod 2$$

Κρυπτοσυστήματα ροής - Τρόποι λειτουργίας

Διακρίνονται σε synchronous (το κλειδί δεν εξαρτάται από το plaintext), και asynchronous (λέγονται και self-synchronizing).

Επίσης σε periodic ($\forall i : z_{i+d} = z_i$, όπου d η περίοδος) και aperiodic.

Παράδειγμα: το Vigenère είναι synchronous και periodic.

Κρυπτοσυστήματα Ροής Γραμμικής Ανάδρασης

Κρυπτοσυστήματα ροής: Linear Recurrence Keystream

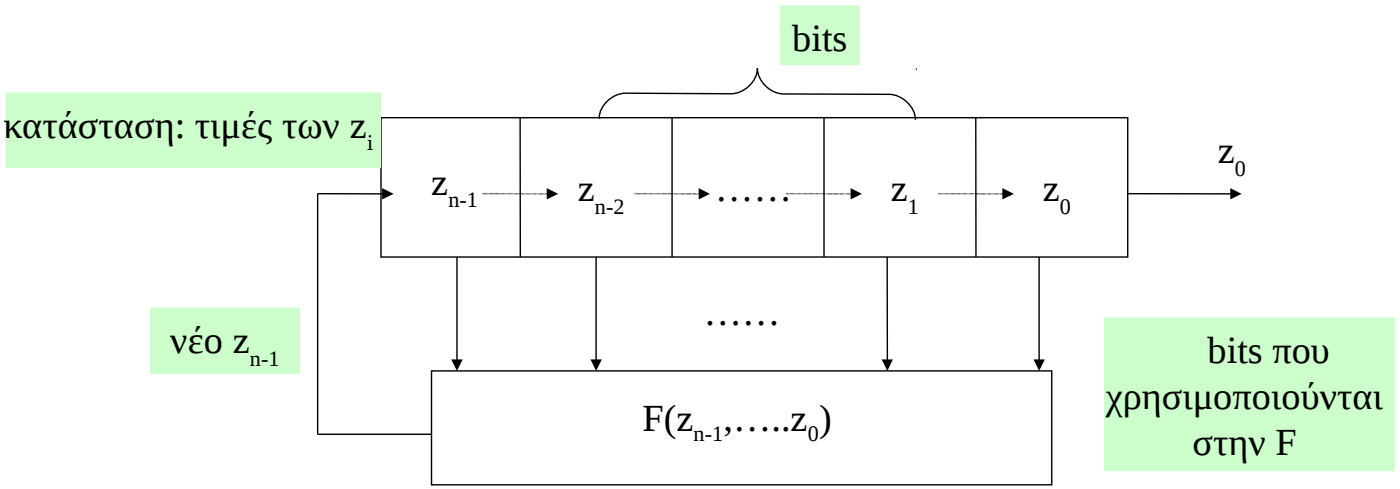
Αρχικό διάνυσμα κλειδιών: $(z_0, z_1, \ldots, z_{m-1})$.
Τα υπόλοιπα κλειδιά υπολογίζονται ως εξής:

$$z_{i+m} = \sum_{j=0}^{m-1} c_j \cdot z_{i+j} \pmod 2, \quad \forall j, c_j \in \{0, 1\}$$

Εάν το πολυώνυμο $c_0 + c_1 x + c_2 x^2 + \ldots + c_{m-1} x^{m-1} + x^m$ είναι primitive, τότε το κρυπτοσύστημα έχει περίοδο $d = 2^m - 1$.
Π.χ. $c_0 = c_1 = 1, c_2 = c_3 = 0$ ορίζουν το πολυώνυμο $x^4 + x + 1$, και με δεδομένο αρχικό κλειδί $z_0, \ldots, z_3$ έχουμε
$z_{4+i} = z_i + z_{i+1} \bmod 2$.

Το κρυπτοσύστημα αυτό έχει περίοδο 15.

Υλοποίηση με Linear Feedback Shift Register (LFSR).


bits
κατάσταση: τιμές των z_i
z_{n-1}
z_{n-2}
.....
z_1
z_0
z_0
.....
νέο z_{n-1}
$F(z_{n-1},.....z_0)$
bits που χρησιμοποιούνται στην F

Σχήμα 5: FSR

Καταχωρητές Ολίσθησης Γραμμικής Ανάδρασης - LFSRs

- Δημιουργούν περιοδικές ακολουθίες, με περίοδο το πολύ $2^L - 1$, όπου L το πλήθος των ψηφίων.
- Αν το αντίστοιχο πολυώνυμο είναι primitive έχουμε maximum-length LFSR. Πολλά γνωστά primitive πολυώνυμα.
- Σημαντικό μέγεθος για ακολουθίες: γραμμική πολυπλοκότητα (linear complexity). Είναι το ελάχιστο μέγεθος LFSR που παράγει την ίδια ακολουθία.
- Αλγόριθμος Berlekamp-Massey: υπολογίζει τη γραμμική πολυπλοκότητα και τον αντίστοιχο LFSR.
- Αύξηση γραμμικής πολυπλοκότητας: χρήση περισσότερων LFSRs, συνδυασμός εξόδων με μη γραμμικό τρόπο.
  Π.χ. Geffe generator συνδυάζει 3 maximum-length LFSRs με μήκος L_1, L_2, L_3 και εξόδους x_1, x_2, x_3:
  $$f(x_1, x_2, x_3) = x_1x_2 \oplus (1 \oplus x_2)x_3$$
  έχει περίοδο $(2^{L_1} - 1) \cdot (2^{L_2} - 1) \cdot (2^{L_3} - 1)$ και γραμμική πολυπλοκότητα $L = L_1L_2 + L_2L_3 + L_3$

Κρυπτοσυστήματα ροής με LFSRs

- LFSR: εύκολη υλοποίηση σε hardware, καλές στατιστικές ιδιότητες, αλλά μη ασφαλή γιατί τα bits εξόδου έχουν γραμμική σχέση
- Αντιμετώπιση
  - μη γραμμική ανάδραση: έξοδος μη γραμμική συνάρτηση των registers
  - συνδυασμός των εξόδων περισσότερων LFSRs, αλλά χωρίς εξάρτηση της τελικής εξόδου από κάποια μεμονωμένη
- Χρήση σε:
  1. DVD (CSS): 2 LFSRs, (ανάκτηση σπόρου σε 2^{17})
  2. GSM (A5/1): 3 LFSRs ($2^{39.91}$, με προεργασία 2^{38}), (A5/2): 4 LFSRs
  3. Bluetooth (E0): 4 LFSRs (ανάκτηση σπόρου σε 2^{38})

Άλλα κρυπτοσυστήματα ροής

Πρόγραμμα eStream

- eStream project: 2004-2008
- Κατηγορίες:
  - Μήκος κλειδιού 128 bits και ένα IV (initialization vector) μήκους 64 και/ή 128 bits (SW)
  - Μήκος κλειδιού 80 bits και ένα IV (initialization vector) μήκους 32 και/ή 64 bits (HW)
- Ξεχωριστές προτάσεις για SW και για HW
- Αξιολόγηση:
  - Ασφάλεια
  - Δωρεάν αδειοδότηση
  - Επιδόσεις και φάσμα εφαρμογών
- Η επιτροπή απλώς συγκέντρωσε τις συμμετοχές, η αξιολόγηση έγινε από την κοινότητα

Πρόγραμμα eStream

- Κριτήρια ασφάλειας
  - οποιαδήποτε επίθεση ανάκτησης κλειδιού πρέπει να είναι τόσο δύσκολη όσο η εξαντλητική αναζήτηση
  - Απλότητα σχεδίασης
- Κριτήρια υλοποίησης
  - SW και HW αποδοτικότητα
  - Εκτέλεση και μνήμη
  - Επίδοση
  - Ευελιξία χρήσης

SW	HW
HC-128	Grain v1
Rabbit	MICKEY 2.0
Salsa20	Trivium
Sosemanuk	

Λεπτομέρειες στο `https://en.wikipedia.org/wiki/ESTREAM`