Το κρυπτοσύστημα RSA

Παναγιώτης Γροντάς - Άρης Παγουρτζής

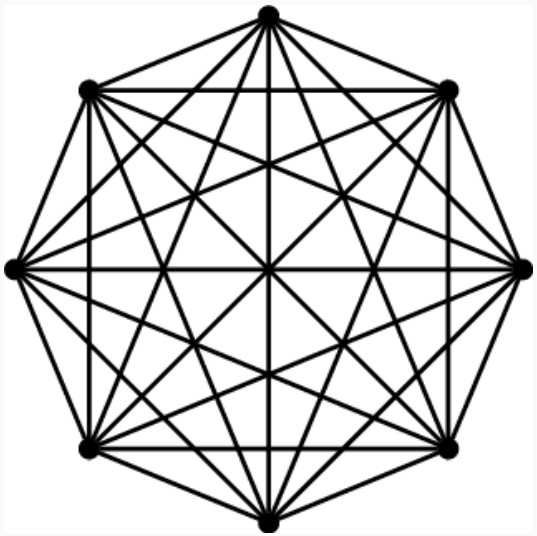
24/11/2023

ΕΜΠ - Κρυπτογραφία (2022-2023)

Περιεχόμενα

Ασύμμετρη Κρυπτογραφία

Συμμετρικά Κρυπτοσυστήματα - Το Μειονέκτημα
Διαχείριση και Διανομή Κλειδιών



Συγκεκριμένα

- Νέα κλειδιά ανά ζεύγος
- Για n χρήστες χρειάζονται $\frac{n(n-1)}{2}$ κλειδιά
- Οι χρήστες πρέπει 'συναντηθούν' για να ανταλλάξουν κλειδιά
- Εύκολο σε ελεγχόμενα περιβάλλοντα, δύσκολο σε ανοικτά
- Δυσκολίες διαχείρισης (πχ. ενημέρωση για έκδοση νέων), αποθήκευσης

Η αρχή της σύγχρονης Κρυπτογραφίας

Η λύση μετά από *2500* χρόνια προσπαθειών:

Whitfield Diffie, Martin Hellman
New Directions in Cryptography - *(1976)*

με σημαντική βοήθεια από: Ralph Merkle

Ίσως και νωρίτερα (σύμφωνα με αποχαρακτηρισμένα έγγραφα):

- James H. Ellis (1970 - GHCQ) - no secret encryption
- Clifford Cocks (1973 - GHCQ) - RSA
- Malcolm J. Williamson (1974 - GHCQ) - Diffie Hellman Key Exchange

Δεν εφαρμόστηκαν λόγω της κλειστής φύσης των στρατιωτικών εφαρμογών

Βασική αναλογία

Κλειδωμένο γραμματοκιβώτιο με σχισμή ή κάλπη

- οποιοσδήποτε μπορεί να εισάγει ένα γράμμα (κρυπτογράφηση - δημόσια λειτουργία)
- για άνοιγμα (αποκρυπτογράφηση) χρειάζεται προσπάθεια από οποιονδήποτε...
- *εκτός* από τον κάτοχο του κλειδιού (αποκρυπτογράφηση - ιδιωτική λειτουργία)

New Directions in Cryptography

3 καινοτόμες ιδέες - 1 κατασκευή

1. **Ανταλλαγή Κλειδιού Diffie - Hellman**
   Δημιουργία κοινού κλειδιού πάνω από δημόσιο - μη ασφαλές κανάλι (online)
2. **Κρυπτογραφία Δημοσίου Κλειδιού**
   - Το κλειδί κρυπτογράφησης είναι δημόσιο
   - Το κλειδί αποκρυπτογράφησης είναι ιδιωτικό
   - *n* χρήστες, *n* ζεύγη κλειδιών - Εύκολη διανομή
3. **Ψηφιακή Υπογραφή**
   - Δημιουργία με ιδιωτικό - Επαλήθευση με δημόσιο κλειδί
   - Ακεραιότητα, Αυθεντικότητα, Μη Αποκήρυξη με ασύμμετρο τρόπο

RSA Υλοποίηση μόνο για το (1)

Trapdoor Functions

Συναρτήσεις μονής κατεύθυνσης
Μία συνάρτηση f λέγεται μονής κατεύθυνσης εάν είναι εύκολο να υπολογιστεί το $f(x)$ δεδομένου του x,

ενώ

ο αντίστροφος υπολογισμός του x δεδομένου του $f(x)$ είναι απρόσιτος.

Trapdoor Functions - Ορισμός
Μια συνάρτηση μονής κατεύθυνσης f για την οποία ο υπολογισμός της f^{-1} είναι *εύκολος* ...

όταν δίνεται μια μυστική πληροφορία (secret trapdoor)

Κριτική ασύμμετρης κρυπτογραφίας

- Επίλυση προβλήματος διανομής κλειδιού σε αυθεντικοποιημένα (authenticated) κανάλια
- Κρυμμένη υπόθεση: Ανήκει το pk σε αυτόν που νομίζουμε ότι ανήκει;
- Μετατροπή προβλήματος διανομής κλειδιού σε πρόβλημα αυθεντικότητας κλειδιού
- Λύση πάλι με ασύμμετρη κρυπτογραφία

- Υπερβολικά αργή - πρόβλημα για μεγάλο όγκο δεδομένων σε μη ισχυρές συσκευές
- Hybrid encryption
  Διανομή συμμετρικού κλειδιού με ασύμμετρη κρυπτογραφία
  Κρυπτογράφηση μηνυμάτων με συμμετρική

Ορισμός RSA

RSA (1977)

- Η πρώτη *κατασκευή* κρυπτοσυστήματος δημοσίου κλειδιού
- Ron Rivest, Adi Shamir, Leonard Adleman
- Πατέντα μέχρι το 2000

Το κρυπτοσύστημα

Δημιουργία Κλειδιών ($\mathsf{KGen}(1^\lambda) = (\mathsf{pk}, \mathsf{sk})$)

- Επιλογή πρώτων p, q $\quad \frac{\lambda}{2}$ bits
- Υπολογισμός $n = p \cdot q$ (λ bits)
- Επιλογή e: $1 < e < \phi(n)$ και $gcd(e, \phi(n)) = 1$
- $d = e^{-1} \pmod{\phi(n)}$ με EGCD
- Επιστροφή $(\mathsf{pk}, \mathsf{sk}) = ((e, n), d)$

$\mathbb{Z}_n^* = \{\, x \in \mathbb{Z}_n \mid gcd(x, n) = 1 \,\}$

Κρυπτογράφηση

- $\mathsf{Enc} : \mathbb{Z}_n^* \to \mathbb{Z}_n^*$ με $\mathsf{Enc}((e, n), m) = m^e \bmod n$

Αποκρυπτογράφηση

- $\mathsf{Dec} : \mathbb{Z}_n^* \to \mathbb{Z}_n^*$ με $\mathsf{Dec}(d, c) = c^d \bmod n$

Θα δούμε ότι μπορεί $m, c \in \mathbb{Z}_n$ και όχι $m, c \in \mathbb{Z}_n^*$

Παράδειγμα

KGen

- $(p, q) = (17, 23)$ $n = 391$
- $\phi(n) = 352$
- $e = 3$
- $d = 235$

$\mathsf{Enc}(3, 158) = 158^3 \bmod 391 = 295$

$\mathsf{Dec}(235, 295) = 295^{235} \bmod 391 = 158$

Ορθότητα

Πρέπει: $\mathrm{Dec}(d, \mathrm{Enc}((e, n), m)) = m, \forall m \in \mathbb{Z}_n^*$

$$\mathrm{Dec}(d, \mathrm{Enc}((e, n), m)) = m^{ed} =$$
$$m^{k\phi(n)+1} = m^{\phi(n)k} \cdot m =$$
$$m \pmod{n}$$

λόγω Θ.Euler και αφού $m \in \mathbb{Z}_n^*$

Παρατηρήσεις RSA

Κωδικοποίηση Μηνύματος

Δεν απαιτείται $m \in \mathbb{Z}_n^*$ για ορθότητα. Ισχύει για κάθε $m \in \mathbb{Z}_n$

Απόδειξη
$m \in \mathbb{Z}_n \Rightarrow gcd(m, n) \neq 1 \Rightarrow gcd(m, n) \in \{p, q\}$

Αν δείξουμε ότι: $m^{ed} = m \pmod{p}$ και $m^{ed} = m \pmod{q}$

από CRT θα έχουμε: $m^{ed} = m \pmod{pq}$

Κωδικοποίηση μηνύματος (συνέχεια)

Περίπτωση $gcd(m, n) = p$

Πράγματι $m^{ed} = m \pmod{p}$ γιατί:

$$m^{ed} = m \pmod{p} \Leftrightarrow (kp)^{ed} = 0 \pmod{p}$$

Επίσης $m^{ed} = m \pmod{q}$ γιατί:

$$m^{ed} = m \cdot m^{ed-1} = m \cdot m^{k\phi(n)} = m \cdot m^{k(p-1)(q-1)} = m \cdot 1 \pmod{q}$$

λόγω του Θ. Fermat που ισχύει στο $\mathbb{Z}_q$

Ομοίως και για $gcd(m, n) = q$

Παράμετρος Ασφάλειας - τιμές

Συστάσεις για χρήση modulus:

- 2048 bits: βραχυχρόνια ασφάλεια ($\approx$ 80 bit AES key)
- 3072bits: μακροχρόνια ασφάλεια ($\approx$ 128 bit AES key)

RSA Factoring Challenge

Παραγοντοποίηση Modulus 768bit (2009)

RSA-768 = 1 230 186 684 530 117 755 130 494 958 384 962 720 772 853 569 595 334 792 197 322 452 151 726 400 507 263 657 518 745 202 199 786 469 389 956 474 942 774 063 845 925 192 557 326 303 453 731 548 268 507 917 026 122 142 913 461 670 429 214 311 602 221 240 479 274 737 794 080 665 351 419 597 459 856 902 143 413 = 33 478 071 698 956 898 786 044 169 848 212 690 817 704 794 983 713 768 568 912 431 388 982 883 793 878 002 287 614 711 652 531 743 087 737 814 467 999 489 $\times$ 36 746 043 666 799 590 428 244 633 799 627 952 632 279 158 164 343 087 642 676 032 283 815 739 666 511 279 233 373 417 143 396 810 270 092 798 736 308 917

Παράμετρος Ασφάλειας - τιμές

Παραγοντοποιήθηκε στις 2/12/2009 μετά από 10^{20} υπολογιστικά βήματα

Διάρκεια υπολογισμού: 2+ ημερολογιακά χρόνια χρησιμοποιώντας παράλληλη επεξεργασία

Εκτίμηση: 2000 χρόνια σε single core system (2.2 GHz AMD Opteron με 2GB RAM)

(Factorization of a 768-bit RSA modulus)

Επίσης έχει παραγοντοποιηθεί modulus 795 bits (2019)

Επιλογή πρώτων

- Επιλογή περιττού x ($\frac{\lambda}{2}$ bits)
- Εφαρμογή Primality test (Miller Rabin)
- Επανάληψη μέχρι να βρεθεί

$\Pr[x \text{ πρώτος }] \approx 2 \cdot \frac{\frac{x}{lnx}}{x} \approx \frac{2}{lnx}$

Δηλ. για $|p| = 1024$: $\frac{2}{ln2^{1024}} = \frac{1}{512ln2} = \frac{1}{356}$

Συστάσεις:

- p, q ίδιου μήκους αλλά όχι πολύ κοντά
- p, q safe primes δηλ. $p-1, q-1$ έχουν μεγάλους πρώτους παράγοντες
- Αλλά και $p+1, q+1$ έχουν μεγάλους πρώτους παράγοντες
- Προσοχή στην τυχαιότητα: επιλογή p, q ομοιόμορφα και ανεξάρτητα

Επιλογή εκθέτη κρυπτογράφησης

Θέλουμε ταχύτατη κρυπτογράφηση

- Εύκολος Υπολογισμός Δύναμης Με Repeated Squaring (Square και Multiply)
  - Αναπαράσταση e στο δυαδικό
  - Για κάθε 0 ύψωση στο τετράγωνο
  - Για κάθε 1 ύψωση στο τετράγωνο και πολλαπλασιασμός
- Τεράστια διαφορά πχ. υπολογισμός x^e για $|e| = 1024$ bits
  Χωρίς repeated squaring: $2^{1024} \approx 10^{300}$ πολλαπλασιασμοί
  Με repeated squaring περίπου $1.5 \cdot 1024 = 1536$ πολλαπλασιασμοί
- Ελαχιστοποίηση Πολλαπλασιασμών: Low Hamming Weight
- Μπορεί e να είναι πρώτος
- Ανεξάρτητη επιλογή από p, q - πάντα ίδιος
- Παράδειγμα: $e \in \{3, 17, 65537 = 2^{16} + 1 \quad (RFC4871)\}$

Βελτίωση αποκρυπτογράφησης

Πρόβλημα: Το κλειδί αποκρυπτογράφησης δεν μπορεί να είναι μικρό

- Επιθέσεις brute force
- Εξειδικευμένες επιθέσεις
- $|d| > \frac{\lambda}{3}$

Επιτάχυνση αποκρυπτογράφησης με 'συνιστώσες' CRT

- Υπολογισμός $c_p = c \bmod p, c_q = c \bmod q$
- Υπολογισμός $d_p = d \bmod (p-1)$, $d_q = d \bmod (q-1)$,
- Υπολογισμός $m_p = c_p^{d_p} \bmod p, m_q = c_q^{d_q} \bmod q$
- Συνδυασμός με CRT για m

Βελτίωση ταχύτητας:4 φορές

Ασφάλεια

Σχετιζόμενα (Δύσκολα) Προβλήματα

Το πρόβλημα RSA (e-οστές ρίζες)
Δίνονται $n = pq$, e με $gcd(e, \phi(n)) = 1$ και $c \in \mathbb{Z}_n^*$. Να βρεθεί η τιμή $c^{\frac{1}{e}(=d)} = m$.

Το πρόβλημα RSA-KINV
Δίνονται $n = pq$, e με $gcd(e, \phi(n)) = 1$. Να βρεθεί η τιμή $e^{-1} \pmod{\phi(n)} (= d)$.

Το πρόβλημα FACTORING
Δίνεται $n = pq$ με p, q πρώτοι. Να βρεθούν τα p, q.

Το πρόβλημα COMPUTE-$\phi(n)$
Δίνεται $n, \phi(n)$ με $n = pq$ όπου p, q πρώτοι. Να βρεθούν τα p, q.

Σχέσεις Προβλημάτων (1)

RSAP $\leq$ RSA-KINV
Αν βρεθεί $d = e^{-1}$ υπολογίζεται εύκολα $c^d \bmod n$

RSA-KINV $\leq$ FACTORING
Έστω ότι μπορούν να βρεθούν p, q για $n = pq$ (λύση FACTORING)

Υπολογισμός $\phi(n) = (p-1) \cdot (q-1)$

Χρήση EGCD για εύρεση e^{-1} (όπως KGen)

Σχέσεις Προβλημάτων (2)

COMPUTE-$\phi(n)$ $\equiv$ FACTORING
Προφανώς: COMPUTE-$\phi(n)$ $\leq$ FACTORING

Αλλά και FACTORING $\leq$ COMPUTE-$\phi(n)$ επειδή:

$n = pq$ και $\phi(n) = (p-1)(q-1)$

$\phi(n) = n - (p+q) + 1 \Rightarrow p + q = n - \phi(n) + 1$

Προκύπτει η εξίσωση $x^2 - (n - \phi(n) + 1)x + n = 0$ από όπου παίρνουμε p, q

FACTORING $\leq^r$ RSA-KINV (RSA, 1977)

Αν γνωρίζουμε τον $d = e^{-1}$ μπορούμε να κατασκευάσουμε αλγόριθμο παραγοντοποίησης του n με βάση τον Miller Rabin

- Υπολογίζουμε $s = ed - 1 = k\phi(n) = 2^t l$ με l μονό
- Επιλέγουμε $a \in \{2, \cdots, n-1\}$ με $gcd(a, n) = 1$
- Από Θ. Euler $a^s = 1 \pmod n$
- Υπολογίζουμε $r_1 = a^{\frac{s}{2}}$. Τότε $r_1^2 - 1 = (r_1 - 1)(r_1 + 1) = 0 \pmod n$
- Αν $r_1 \neq \pm 1 \pmod n$ τότε $(p, q) = (gcd(r_1 - 1, n), gcd(r_1 + 1, n))$
- Αλλιώς επαναλαμβάνουμε $r_2 = a^{\frac{s}{4}}, r_3 = a^{\frac{s}{8}}, \ldots$
- μέχρι να βρούμε $r_i \neq \pm 1 \pmod n$ (παραγοντοποιήσαμε το n) ή
- το πολύ t φορές ($\frac{s}{2^t} \neq 0 \pmod 2$). Τότε επαναλαμβάνουμε με άλλο a
- Από θεωρία αριθμών:
  - $\Pr[success_a] \geq \frac{1}{2}$
  - m επαναλήψεις $\Pr[success] \geq 1 - \frac{1}{2^m}$

FACTORING $\leq^r$ RSA-KINV (2)

Παράδειγμα:

$$n = 1441499, \quad e = 17, \quad d = 507905$$

$$s = e \cdot d - 1 = 8634385, a = 2$$

$$r_1 = a^{\frac{s}{2}} = 2^{4317192} = 1 \pmod{n}$$

$$r_2 = a^{\frac{s}{4}} = 2^{2158596} = 1 \pmod{n}$$

$$r_3 = a^{\frac{s}{8}} = 2^{1079298} = 119533 \pmod{n}$$

$$p = gcd(r_3 - 1, n) = gcd(119532, 1441499) = 1423$$

$$q = gcd(r_3 + 1, n) = n/p = gcd(119534, 1441499) = 1013$$

Συνολική Εικόνα Προβλημάτων σχετικών με RSA

$\text{RSAP} \leq \text{RSA-KINV} \leq \text{COMPUTE-}\phi(N) \equiv \text{FACTORING} \leq^r \text{RSA-KINV}$

Αργότερα (May, 2004) $\text{FACTORING} \leq \text{RSA-KINV}$

Τελικά:

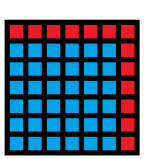
$\text{RSAP} \leq \text{RSA-KINV} \equiv \text{COMPUTE-}\phi(N) \equiv \text{FACTORING}$

Το RSAP λοιπόν δεν είναι δυσκολότερο από το FACTORING

Μάλλον είναι ευκολότερο αλλά δεν γνωρίζουμε ακριβώς πόσο.
Υπόθεση RSA: Το RSAP είναι υπολογιστικά απρόσιτο.

Επιθέσεις

Παραγοντοποίηση Fermat για κοντινά p, q



- Επιλογή κοντινών $p, q \approx \sqrt{n}$ επιτρέπει την παραγοντοποίηση του n
- $\exists a, b : n = a^2 - b^2 = (a + b)(a - b) = p \cdot q$
- Άρα $b^2 = a^2 - n$ (ξέρουμε n)
- Αρχίζουμε να μαντεύουμε a από $a = \sqrt{n}$
- Υπολογίζουμε b^2 και ελέγχουμε αν είναι τετράγωνο
- Επαναλαμβάνουμε με $a = a + 1$

Δυστυχώς πολλές υλοποιήσεις ακόμα και σήμερα είναι ευάλωτες (*https://fermatattack.secvuln.info/*)

Επίθεση κοινού γινομένου

Ιδέα
Χρήση κοινού n για να μειωθεί το κόστος πράξεων modulo

Σενάριο
KDC διαθέτει $n = pq$ και μοιράζει στους χρήστες A, B τα κλειδιά (e_A, d_A) και (e_B, d_B).

Εσωτερική Επίθεση (από γνώστη του d_A)

- Ο A αφού γνωρίζει το d_A μπορεί να παραγοντοποιήσει το n (αναγωγή FACTORING $\leq^r$ RSA-KINV)
- Υπολογισμός $\phi(n)$
- Ευρεση $d_B = e_B^{-1} \pmod{\phi(n)}$ με EGCD
- Διάβασμα όλων των μηνυμάτων του B

Επίθεση κοινού γινομένου - παραλλαγή

Εξωτερική Επίθεση

- Ο $\mathcal{A}$ γνωρίζει $(n, e_1), (n, e_2)$
- Μπορεί να ανακτήσει οποιοδήποτε m κρυπτογραφηθεί και με τα δύο δημόσια κλειδιά
- Δηλ. ο $\mathcal{A}$ διαθέτει c_1, c_2 με
  - $c_1 = m^{e_1} \bmod n$
  - $c_2 = m^{e_2} \bmod n$
- Αν $gcd(e_1, e_2) = 1$ (πολύ πιθανό) τότε με τον EGCD μπορούν να βρεθούν αποδοτικά t_1, t_2:

$$e_1 t_1 + e_2 t_2 = 1$$

- $c_1^{t_1} c_2^{t_2} = m^{e_1 t_1} m^{e_2 t_2} = m^1 = m$

Ron was wrong, Whit is right (2012)

- Συλλογή δημοσίων κλειδιών (e_i, N_i)
- Υπολογισμός $gcd(N_i, N_j) \forall (i, j)$
- Αν $gcd(N_i, N_j) \neq 1$ τότε (N_i, N_j) μπορούν να παραγοντοποιηθούν
- 0.2% πραγματικών δημοσίων κλειδιών έχουν κοινό πρώτο
- Μάλλον οφείλεται σε πρόβλημα γεννήτριας τυχαίων πρώτων

Επίθεση μικρού δημόσιου εκθέτη

Κακή ιδέα
Χρήση $e = 3$ για να μειωθεί το κόστος κρυπτογράφησης

- Τρία δημόσια κλειδιά $k_1 = (3, n_1), k_2 = (3, n_2), k_3 = (3, n_3)$
- Ο $\mathcal{A}$ γνωρίζει 3 κρυπτογραφήσεις του μηνύματος-στόχου m
  - $c_1 = \mathsf{Enc}(k_1, m) = m^3 \bmod n_1$
  - $c_2 = \mathsf{Enc}(k_2, m) = m^3 \bmod n_2$
  - $c_3 = \mathsf{Enc}(k_3, m) = m^3 \bmod n_3$
- Χρήση CRT για υπολογισμό του $m^3 \bmod n_1 n_2 n_3$
- Αλλά $m^3 < n_1 n_2 n_3$ αφού $m < n_1$ και $m < n_2$ και $m < n_3$
- Εύρεση μηνύματος ως $m = \sqrt[3]{m^3}$

Παρατήρηση: Η επίθεση γενικεύεται και για $e > 3$ (Coppersmith)

Επίθεση μικρού ιδιωτικού εκθέτη - Θεωρία

Αναπαράσταση Με Συνεχή Κλάσματα
Έστω $x \in \mathbb{R}$. Τότε $\exists a_0, a_1, a_2, a_3, \cdots : x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cfrac{1}{\cdots}}}}$

Η αναπαράσταση συμβολίζεται ως $[\alpha_0, \alpha_1, \cdots]$

υποσύνολο όρων - προσέγγιση x

Αν $x \in \mathbb{Q}$ τότε η αναπαράσταση είναι πεπερασμένη

Μπορεί να υπολογιστεί με EGCD.

Θεώρημα
Αν gcd(a,b) = 1 και $|x - \frac{a}{b}| < \frac{1}{2b^2}$ τότε το κλάσμα $\frac{a}{b}$ εμφανίζεται στην προσέγγιση με συνεχή κλάσματα του x.

Βασική ιδέα επίθεσης
Για μικρές τιμές του d ($3d < n^{\frac{1}{4}}$) μπορούμε να βρούμε το d μέσω της αναπαράστασης με συνεχή κλάσματα αν $q < p < 2q$.

Επίθεση μικρού ιδιωτικού εκθέτη - Προσαρμογή (1)

Αφου $n = pq > q^2$ ισχύει $q < \sqrt{n}$

$$n - \phi(n) = pq - (p-1)(q-1) = p + q - 1 < 2q + q < 3\sqrt{n} \tag{1}$$

Ο $\mathcal{A}$ γνωρίζει το e και ότι $\exists k : ed = 1 + k\phi(n)$ Επίσης ισχύει

$$e < \phi(n) \Rightarrow ke < k\phi(n) < 1 + k\phi(n) = ed \Rightarrow k < d \tag{2}$$

Επίσης:

$$|\frac{e}{n} - \frac{k}{d}| = |\frac{ed - kn}{dn}| = |\frac{1 + k\phi(n) - kn}{dn}| = |\frac{1 - k(n - \phi(n))}{dn}| \leq \frac{1 + k(n - \phi(n))}{dn}$$

Επίθεση μικρού ιδιωτικού εκθέτη - Προσαρμογή (2)

Από την σχέση (1):

$$|\frac{e}{n} - \frac{k}{d}| < \frac{3k\sqrt{n}}{dn} = \frac{3k}{d\sqrt{n}}$$

Από την σχέση (2):

$$|\frac{e}{n} - \frac{k}{d}| < \frac{3}{\sqrt{n}}$$

Από την υπόθεση για το μέγεθος του d έχουμε:

$$d < \frac{\sqrt[4]{n}}{3} \Rightarrow d^2 < \frac{\sqrt{n}}{9} \Rightarrow 2d^2 < \frac{2\sqrt{n}}{9} < \frac{\sqrt{n}}{3} \Rightarrow \frac{3}{\sqrt{n}} < \frac{1}{2d^2}$$

Επίθεση μικρού ιδιωτικού εκθέτη - Προσαρμογή (3)

Τελικά:

$$|\frac{e}{n} - \frac{k}{d}| < \frac{1}{2d^2}$$

Επειδή $gcd(k, d) = 1$ το κλάσμα k/d εμφανίζεται στην προσέγγιση του e/n με συνεχή κλάσματα.

Πώς μπορεί να το εκμεταλλευτεί ο αντίπαλος $\mathcal{A}$;

Επίθεση μικρού ιδιωτικού εκθέτη

Διαδικασία

- Επιλογή μηνύματος m από τον $\mathcal{A}$ και κρυπτογράφηση σε c
- Κατασκευή αναπαράστασης του e/n με συνεχή κλάσματα
- Ύψωση c σε κάθε έναν από τους παρονομαστές της
- Ο παρονομαστής που επιτυγχάνει σωστή αποκρυπτογράφηση είναι το d

Επίθεση μικρού ιδιωτικού εκθέτη - Παράδειγμα

$(e, n) = (207031, 242537)$

Προσεγγίσεις-δοκιμές για $m = 8$ και $c = 46578 = 8^{207031} \bmod 242537$

$$\frac{207031}{242537} = 0 + \frac{1}{\frac{242537}{207031}} = 0 + \frac{1}{1 + \frac{35006}{207031}} = 0 + \frac{1}{1 + \frac{1}{\frac{207031}{35006}}} = 0 + \frac{1}{1 + \frac{1}{5 + \frac{32280}{35006}}} = 0 + \frac{1}{1 + \frac{1}{5 + \frac{1}{1 + \frac{35006}{32280}}}}} = \cdots$$

$$[0; 1] = 0 + \frac{1}{1} = 1 \qquad \text{και}$$

$$46578^1 \bmod 242537 = 46578$$

$$[0; 1; 5] = 0 + \frac{1}{1 + \frac{1}{5}} = \frac{5}{6} \qquad \text{και}$$

$$46578^6 \bmod 242537 = 175938$$

$$[0; 1; 5; 1] = 0 + \frac{1}{1 + \frac{1}{5+1}} = \frac{6}{7} \qquad \text{και}$$

$$46578^7 \bmod 242537 = 8$$

Άρα $d = 7$

Side Channel Attacks

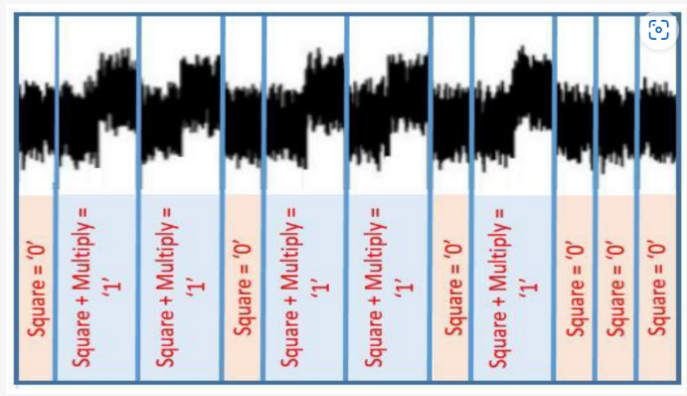
Φυσική πληροφορία που διαρρέει από την επεξεργασία κρυπτοκειμένου - κλειδιού

- Κατανάλωση ρεύματος
- Αυξομειώσεις συχνότητας επεξεργασίας
- Χρόνος απάντησης

Απαιτείται πρόσβαση στη συσκευή (όχι πάντα)

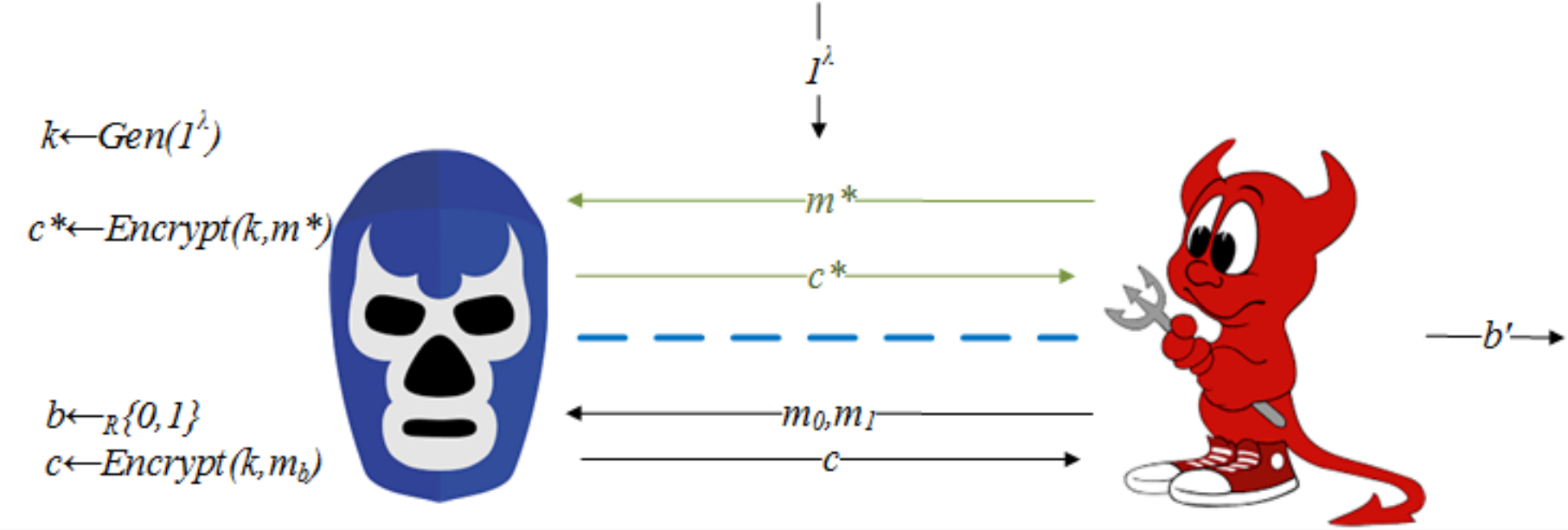
Μπορεί να αποκαλυφθεί ολόκληρο το κλειδί

Λύση: Dummy πράξεις



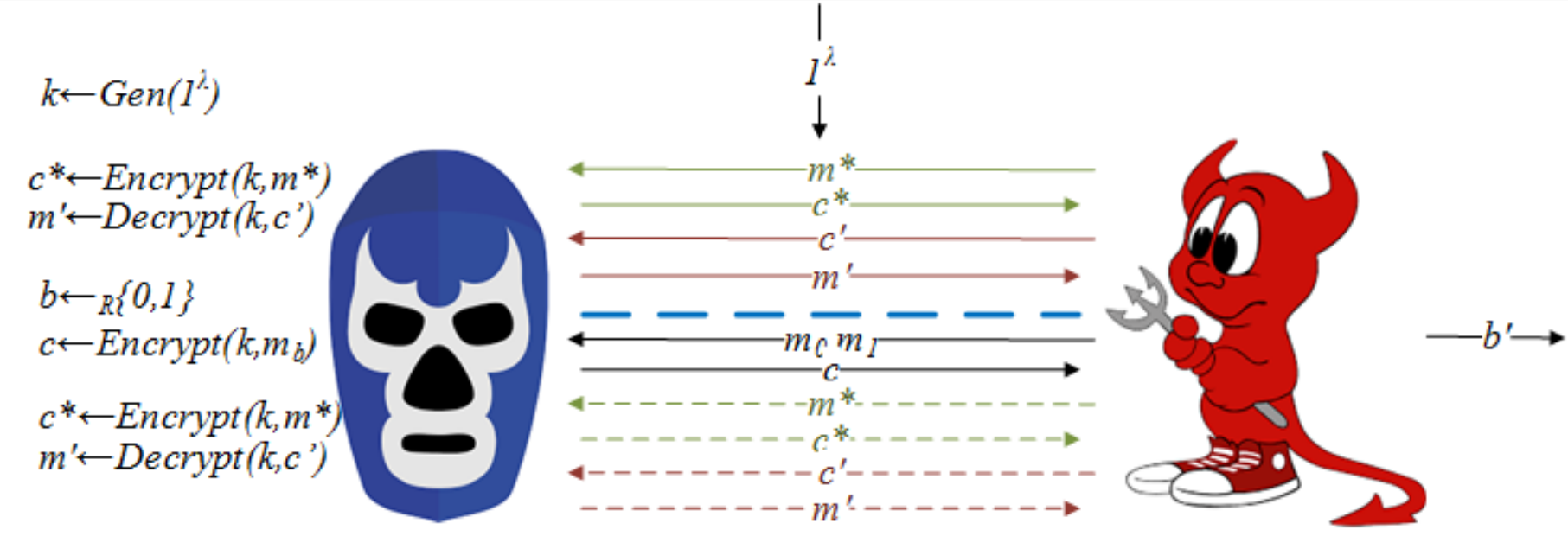
Μοντελοποίηση

Το (textbook) RSA δεν διαθέτει IND-CPA



- Γιατί είναι ντετερμινιστικό
- Ο $\mathcal{A}$ μπορεί να ξεχωρίσει κρυπτογραφήσεις μηνυμάτων
- τις οποίες μπορεί να παράγει ο ίδιος (δημόσιο κλειδί)

Το RSA δεν διαθέτει IND-CCA



Αφού δεν διαθέτει IND-CPA (δεν χρειάζεται το decryption oracle)

Το RSA είναι malleable

Πολλαπλασιαστικός ομομορφισμός
$\text{Enc}((e,n),m_1)\cdot\text{Enc}((e,n),m_2) = m_1^e \bmod n \cdot m_2^e \bmod n = (m_1\cdot m_2)^e \bmod n = \text{Enc}((e,n),m_1\cdot m_2)$

Στο παίγνιο CCA:

- Στόχος: Αποκρυπτογράφηση του $c_b = m_b^e \bmod n$
- Μπορεί να αποκρυπτογραφήσει το $c' = c_b \cdot x^e \bmod n$ όπου το x είναι δικής του επιλογής
- Ανακτά το $m_b = \frac{m'}{x}$
- Αν $m_b = m_0$ επιστρέφει $b^* = 0$ αλλιώς επιστρέφει $b^* = 1$

Διαρροή Πληροφοριών

Τι διαρρέει (χωρίς συνέπειες)

Jacobi symbol $(\frac{c}{n}) = (\frac{m^e}{n}) = (\frac{m^e}{p})(\frac{m^e}{q}) = (\frac{m}{p})(\frac{m}{q}) = (\frac{m}{n})$

Τι διαρρέει (με συνέπειες)

Έστω $c = m^e \bmod n$

$parity((e, n), c) = m \bmod 2$ - τελευταίο bit (LSB) του plaintext

$loc((e, n), c) = m > \frac{n}{2}$ - plaintext κάτω μισό / πάνω μισό του $\mathbb{Z}_n$

Θεώρημα (Goldwasser, Micali, Tong)

Θεώρημα
Για κάθε στιγμιότυπο του RSA (e,n), και $c = m^e \bmod n$ τα παρακάτω είναι ισοδύναμα:

1. Υπάρχει ένας αποδοτικός αλγόριθμος $\mathcal{A}$ τέτοιος ώστε $\mathcal{A}(c) = m, \forall m \in \mathbb{Z}_n$
2. Υπάρχει ένας αποδοτικός αλγόριθμος που υπολογίζει την συνάρτηση *parity*
3. Υπάρχει ένας αποδοτικός αλγόριθμος που υπολογίζει την συνάρτηση *loc*

Θεώρημα (Goldwasser, Micali, Tong)

Θα δείξουμε ότι: $loc(c) = parity(c \cdot \mathrm{Enc}(2))$

Έχουμε: $parity(c \cdot \mathrm{Enc}(2)) = parity(\mathrm{Enc}(2 \cdot m)) = (2m \bmod n) \bmod 2$

$loc(c) = 1 \Rightarrow m > \frac{n}{2} \Rightarrow 2m > n$

Αφού $n < 2m < 2n$: $2m \bmod n = 2m - n$

Αφού n μονός θα είναι και $2m - n$ μονός δηλ. $parity(c \cdot \mathrm{Enc}(2)) = 1$

$loc(c) = 0 \Rightarrow m \leq \frac{n}{2} \Rightarrow 2m \leq n$

Άρα $(2m \bmod n) \bmod 2 = 0$ δηλ. $parity(c \cdot \mathrm{Enc}(2)) = 0$

Αυτό σημαίνει και: $parity(c) = loc(c \cdot \mathrm{Enc}(2^{-1}))$

Θεώρημα (Goldwasser, Micali, Tong)

Προφανώς $(1) \Rightarrow (3)$ (αν μπορώ να αποκρυπτογραφήσω ξέρω *loc*)

Για το $(3) \Rightarrow (1)$

Δυαδική αναζήτηση, για το m, χρησιμοποιώντας την *loc* και διαδοχικές 'ολισθήσεις' προς τα αριστερά:

$loc(\mathrm{Enc}(m)) = 0 \iff m \in [0, \frac{n}{2})$ και

$loc(\mathrm{Enc}(2m)) = 0 \iff m \in [0, \frac{n}{4}) \cup (\frac{n}{2}, \frac{3n}{4})$

$loc(\mathrm{Enc}(4m)) = 0 \iff m \in [0, \frac{n}{8}) \cup (\frac{n}{2}, \frac{5n}{8})$

Άρα αν $loc(\mathrm{Enc}(m)) = 0$ και $loc(\mathrm{Enc}(2m)) = 0$ και $loc(\mathrm{Enc}(4m)) = 0$ τότε $m \in [0, \frac{n}{8})$

$\ldots$ για $\log_2 n$ βήματα.

Παραλλαγές RSA με IND-CPA

padded-RSA

Βασική ιδέα

- Προσθήκη ψηφίων τυχαιοποίησης r στο μήνυμα.
- Κρυπτογράφηση $f(m, r)$
- Αποκρυπτογράφηση
- Αντιστροφή f (πρέπει να γίνεται εύκολα)

pkcs1 v1.5

$f(m, r) = r||m$ και $|m| = l$.

- Πριν την κρυπτογράφηση δημιουργείται το μήνυμα: $\bar{m} = r||m$, όπου r είναι μια τυχαία συμβολοσειρά από $\lambda - l$ bits.
- Μετατροπή του $\bar{m}$ σε ακέραιο
- Η κρυπτογράφηση γίνεται (κανονικά) ως: $\bar{c} = \bar{m}^e \bmod n$
- Η αποκρυπτογράφηση γίνεται (κανονικά) ως $\bar{c}^d \bmod n = \bar{m}$
- Από το $\bar{m}$ κράταμε μόνο τα l bits χαμηλότερης τάξης.

Αποδεικνύεται ότι διαθέτει ασφάλεια IND-CPA, όχι όμως IND-CCA (μπορούμε να εκμεταλλευτούμε την δομή του padded plaintext)

Padding Oracle Attacks

Βασική Ιδέα: Padding Oracle

Χρήση ενός συστήματος το οποίο μπορεί να αποφανθεί αν ένα κρυπτοκείμενο έχει προκύψει με σωστό padding

Γενική μορφή

- Κατά την αποκρυπτογράφηση, ο παραλήπτης ελέγχει αν το μήνυμα εχει το σωστό padding
- Αν ναι, το επεξεργάζεται
- Αν όχι, το απορρίπτει
- Ο αποστολέας μπορεί να καταλάβει τον λόγο απόρριψης μέσω:
  - ειδικού μηνύματος λάθους
  - side channel: χρόνος απάντησης
- Η επίθεση:
  - Τροποποιούμε ένα ciphertext (CCA) επαναληπτικά
  - Ώστε να πάρουμε πληροφορίες για τη δομή του μέσω του padding
  - και τελικά να μάθουμε (κάτι) για το ciphertext

Δεν αφορά μόνο το RSA!

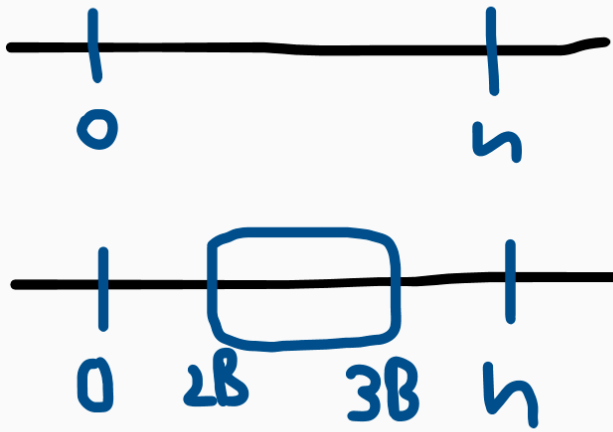
Η επίθεση του Bleichenbacher (Million Message Attack) - I

- Ακριβής Μορφή padded μηνύματος στο pkcs1:
  $PKCS(r, m) = 0\text{x}\ \ \boldsymbol{00}||\boldsymbol{02}||\boldsymbol{r}||\boldsymbol{00}||\boldsymbol{m}$
- Μετά την αποκρυπτογράφηση:
  - Έλεγχος πρώτου byte για την τιμή 0
  - Έλεγχος δεύτερου byte για την τιμή 2
  - Τυχαιότητα r
  - Αναζήτηση του byte 0
  - Ανάκτηση του m

Η επίθεση του Bleichenbacher (Million Message Attack) - II

Η επίθεση:

- Στόχος: Αποκρυπτογράφηση ενός c
- Μορφή $c = PKCS(r, m)^e \bmod n$
- $| PKCS(r, m) | = \lambda$
- Ξεκινάει με $0x0002$ (τα πρώτα 2 bytes)
- Ως αριθμός:
  - $2 \cdot 2^{\lambda-16} \leq PKCS(r, m) < 3 \cdot 2^{\lambda-16}$
  - $2B \leq PKCS(r, m) < 3B$
  - με $B = 2^{\lambda-16}$



Η επίθεση του Bleichenbacher (Million Message Attack) - III

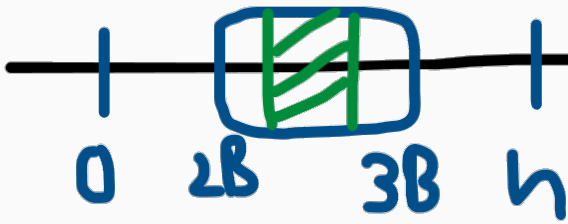
Ο $\mathcal{A}$:

- Διαλέγει *πολλά* τυχαία $s \in \mathbb{Z}_n$
- Φτιάχνει ciphertexts της επιλογής του (CCA):
  $c' = s^e c \bmod n$
- Ομομορφισμός RSA:
  $c' = (sPKCS(r, m))^e \bmod n$
- Τα στέλνει στο padding oracle
- Για τα περισσότερα: λάθος padding.

Η επίθεση του Bleichenbacher (Million Message Attack) - IV

Αν *δεν* δώσει σφάλμα padding:

- Ξέρουμε ότι το padded plaintext έχει σωστή μορφή δηλαδή:
  $\exists r', m' : c' = (PKCS(r', m'))^e \pmod n$
- Αφού $PKCS(r', m') = sPKCS(r, m) \pmod n$ τότε:
  - $2B \leq sPKCS(r, m) - kn \leq 3B - 1$
  - $\frac{2B+kn}{s} \leq PKCS(r, m) < \frac{3B+kn}{s}$
  - και $2B \leq PKCS(r, m) < 3B$
  - Παίρνουμε πληροφορία για k
  - Επαναλαμβάνουμε για νέα s
  - Μέχρι να βρεθεί μια τιμή για το $PKCS(r, m)$



Με 300.000 εως 2.000.000 c' μπορεί να βρεθεί το m ($\lambda = 2048$)

Η επίθεση του Bleichenbacher (Million Message Attack) - IV

Λύσεις

- Αφαίρεση μηνύματος λάθους για padding
- Τροποποίηση ώστε να υπάρχει ασφάλεια IND-CCA2

Δυστυχώς η επίθεση ισχύει ακόμα: The R.O.B.O.T. attack

RSA-OAEP (pkcs1 v2.0) i

Βασική Ιδέα

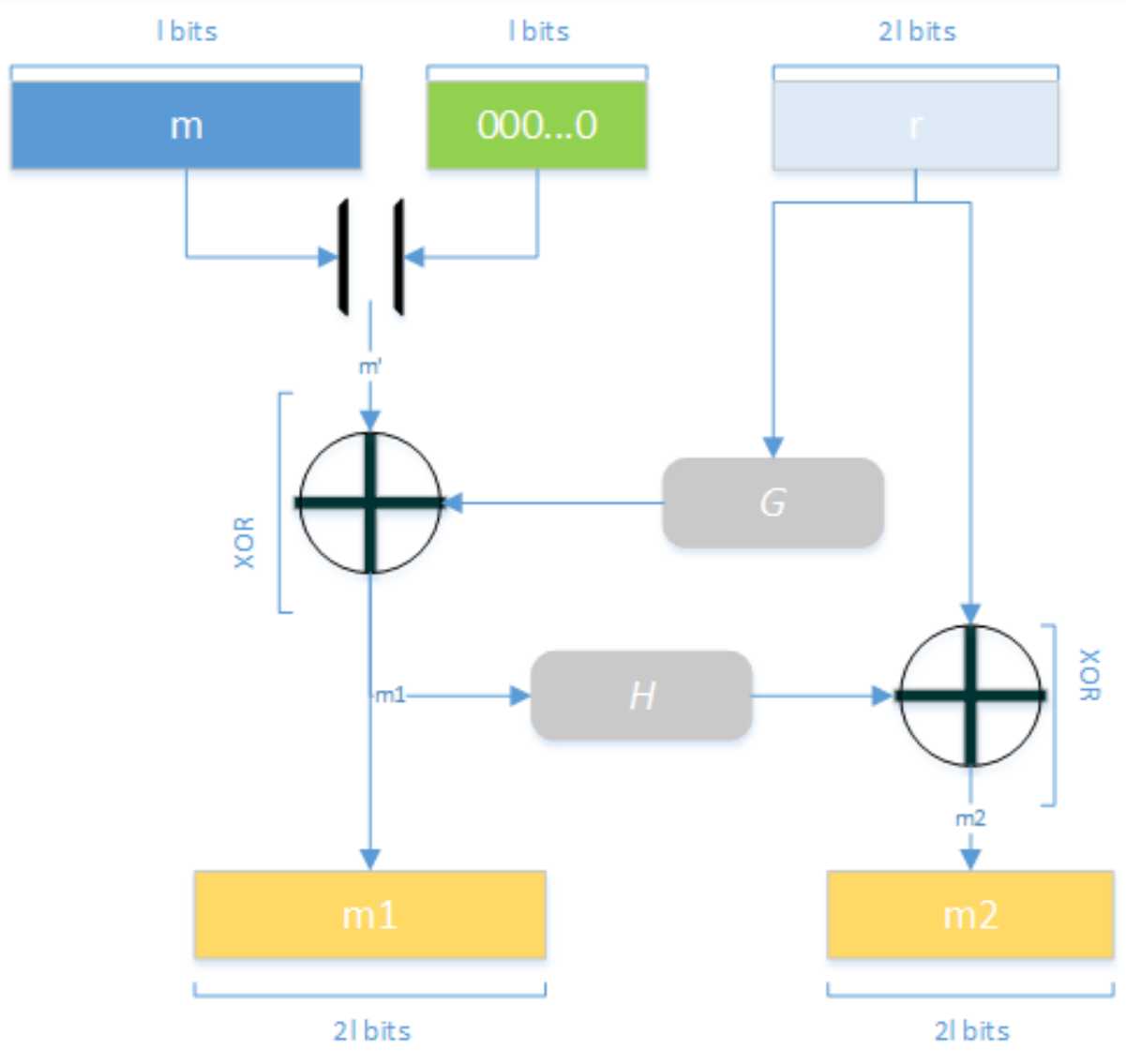
Τα τυχαία bits πρέπει να 'διαχυθούν' σε όλο το κρυπτοκείμενο (Δίκτυα Feistel)

Πρέπει να υπάρχει κάποιου είδους δέσμευση στο αρχικό μήνυμα ενσωματωμένη στο κρυπτοκείμενο (Συνάρτηση Σύνοψης)

Υποθέσεις

- $|m| = l$
- $\mathcal{G}, \mathcal{H} : \{0,1\}^{2l} \to \{0,1\}^{2l}$ συναρτήσεις σύνοψης
- $r \in \{0,1\}^{2l}$

RSA-OAEP (pkcs1 v2.0) ii



RSA-OAEP (pkcs1 v2.0) iii

Κρυπτογράφηση

- Padding για μέγεθος $2l$: $m' = m||0^l$
- Επιλογή τυχαιότητας r μεγέθους $2l$
- Διάχυση bits τυχαιότητας $m_1 = \mathcal{G}(r) \oplus m'$
- Δέσμευση $m_2 = r \oplus \mathcal{H}(m_1)$
- Συνδυασμός $\bar{m} = m_1||m_2$
- Κρυπτογράφηση $\bar{c} = \bar{m}^e \bmod n$

RSA-OAEP (pkcs1 v2.0) iv

Αποκρυπτογράφηση

- Αποκρυπτογράφηση $\bar{c}^d \bmod n = \bar{m}$
- Θεωρούμε ότι $\bar{m} = m_1||m_2$ (χωρισμός στα δύο)
- $\mathcal{H}(m_1) \oplus m_2$
- Ανακτούμε το r (ιδιότητες XOR)
- $m_1 \oplus \mathcal{G}(r)$
- Ανακτούμε το m'
- Έλεγχος l bits χαμηλότερης τάξης
- Αν είναι 0 τότε ανάκτηση μηνύματος από τα l bits υψηλότερης τάξης