Το κρυπτοσύστημα RSA

Παναγιώτης Γροντάς - Άρης Παγουρτζής

20/11/2018

ΕΜΠ - Κρυπτογραφία (2018-2019)

Περιεχόμενα

Ασύμμετρη Κρυπτογραφία

Συμμετρικά Κρυπτοσυστήματα - Το Μειονέκτημα
Διανομή Κλειδιών

Συγκεκριμένα

- Οι χρήστες πρέπει 'συναντηθούν' για να ανταλλάξουν κλειδιά
- Σε περιβάλλοντα πολλών χρηστών: Ανταλλαγή κλειδιών ανά ζεύγος
- Για n χρήστες χρειάζονται $\frac{n(n-1)}{2}$ κλειδιά
- Εύκολο σε ελεγχόμενα περιβάλλοντα, δύσκολο σε ανοικτά
- Δυσκολίες διαχείρισης (πχ. 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
- για άνοιγμα χρειάζεται προσπάθεια από οποιονδήποτε
- εκτός από τον κάτοχο του κλειδιού

New Directions in Cryptography

3 καινοτόμες ιδέες - 1 κατασκευή

1. **Ανταλλαγή Κλειδιού Diffie - Hellman**
2. **Κρυπτογραφία Δημοσίου Κλειδιού**
   - Το κλειδί κρυπτογράφησης μπορεί να είναι δημόσιο
   - Το κλειδί αποκρυπτογράφησης πρέπει να είναι μυστικό
   - *n* χρήστες, *n* ζεύγη κλειδιών - Εύκολη διανομή
3. **Ψηφιακή Υπογραφή**
   - Ασύμμετρα MACs
   - Δημιουργία με ιδιωτικό - Επαλήθευση με δημόσιο κλειδί
   - Αυθεντικότητα, Μη Αποκήρυξη

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) k

Ορισμός RSA

RSA (1978)

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

Το κρυπτοσύστημα

Δημιουργία Κλειδιών:

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

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 $\mathsf{Dec}(d, c) = c^d \bmod n$

Ορθότητα

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

$$\begin{aligned}
\mathsf{Dec}(d, \mathsf{Enc}((e,n),m)) &= \\
(m^e)^d \bmod n &= \\
m^{ed} \bmod n &= \\
m^{k\phi(n)+1} \bmod n &= \\
m^{\phi(n)k} \cdot m \bmod n &= \\
m \bmod n &
\end{aligned}$$

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

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

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

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:

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

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

Ομοίως και για $gcd(m, n) = q$

Παράμετρος Ασφάλειας - RSA challenge

Παραγοντοποίηση Modulus 768bit

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)

Συστάσεις για χρήση modulus:

- 1024bits: βραχυχρόνια ασφάλεια ($\approx$ 80 bit AES key)
- 2048bits, 3072bits: μακροχρόνια ασφάλεια ($\approx$ 128 bit AES key)

Επιλογή πρώτων

Τυχαία επιλογή ακέραιου $\frac{\lambda}{2}$ bits και εφαρμογή Primality test (Miller Rabin) μέχρι να βρεθεί πρώτος

Συστάσεις:

- p, q ίδιου μήκους
- p, q safe primes δηλ. $p-1, q-1$ έχουν μεγάλους πρώτους παράγοντες
- Αλλά και $p+1, q+1$ έχουν μεγάλους πρώτους παράγοντες

Επιλογή εκθέτη κρυπτογράφησης

Θέλουμε ταχύτατη κρυπτογράφηση

- Εύκολος Υπολογισμός Δύναμης Με Square και Multiply
  - Αναπαράσταση e στο δυαδικό
  - Για κάθε 0 ύψωση στο τετράγωνο
  - Για κάθε 1 ύψωση στο τετράγωνο και πολλαπλασιασμός
- Ελαχιστοποίηση Πολλαπλασιασμών: Low Hamming Weight
- Παράδειγμα: $e \in \{3, 17, 65537 = 2^{16} + 1 (RFC4871)\}$
- Μπορεί e να είναι πρώτος
- Ανεξάρτητη επιλογή από p, q

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 $c^{\frac{1}{e}(=d)}$

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 $\frac{1}{e}$ (όπως KeyGen)

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

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

FACTORING $\leq^r$ RSA-KINV (RSA,1977)
Αν γνωρίζουμε τον $d = e^{-1}$ μπορούμε να κατασκευάσουμε πιθανοτικό αλγόριθμο παραγοντοποίησης του n με βάση τον Miller Rabin

FACTORING $\leq^r$ RSA-KINV

- Υπολογίζουμε $s = ed - 1$
- s είναι ζυγός, άρα $s = 2^t \cdot r$ με $t \geq 1$ και r μονό
- Επιλέγουμε τυχαίο $a \in \{2, \cdots, n-1\}$
- Δύο περιπτώσεις:
  - $gcd(a, n) > 1$: Βρέθηκε - Τερματισμός
  - $gcd(a, n) = 1$: Από Θ. Euler $a^{s(=k\phi(n))} = 1 \pmod{n}$
- Ξεκινάμε από $v = a^r$ και υπολογίζουμε $v^2, v^4, \cdots$
- ... μέχρι να βρούμε $v^{2k} = 1$ και $v^k \neq \pm 1$
- Τότε υπολογίζουμε p, q από $gcd(v^k \pm 1, n)$

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

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 $c = m^3 \bmod n_1 n_2 n_3$
- Αλλά $m^3 < n_1 n_2 n_3$ αφού $m < n_1$ και $m < n_2$ και $m < n_3$
- Εύρεση μηνύματος ως $m = \sqrt[3]{c}$

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 $[\alpha_0, \alpha_1, \cdots]$

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

-προσέγγιση με υποσύνολο όρων

Θεώρημα

Έστω $x \in \mathbb{R}$. Αν $|x - \frac{a}{b}| < \frac{1}{2b^2}$ τότε το κλάσμα $\frac{a}{b}$ εμφανίζεται στην προσέγγιση με συνεχή κλάσματα του x.

Βασική ιδέα

Για μεγάλες τιμές του e (μικρές τιμές του d - $d < \frac{1}{3}n^{\frac{1}{4}}$) μπορούμε να βρούμε το d μέσω της αναπαράστασης με συνεχή κλάσματα.

Επίθεση μικρού ιδιωτικού εκθέτη - Προσαρμογή (1)

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

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

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

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 είναι απλοποιημένο, και κατά συνέπεια θα 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
- 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 $c = 46578 = 8^{207031} \bmod 242537$

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

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

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

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

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

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

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

Άρα $d = 7$

Επίθεση κοινού γινομένου

Πολύ Κακή ιδέα
Χρήση κοινού n για να μειωθεί το κόστος πράξεων modulo

Σενάριο
TTP 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:

$$e_1 t_1 + e_2 t_2 = 1$$

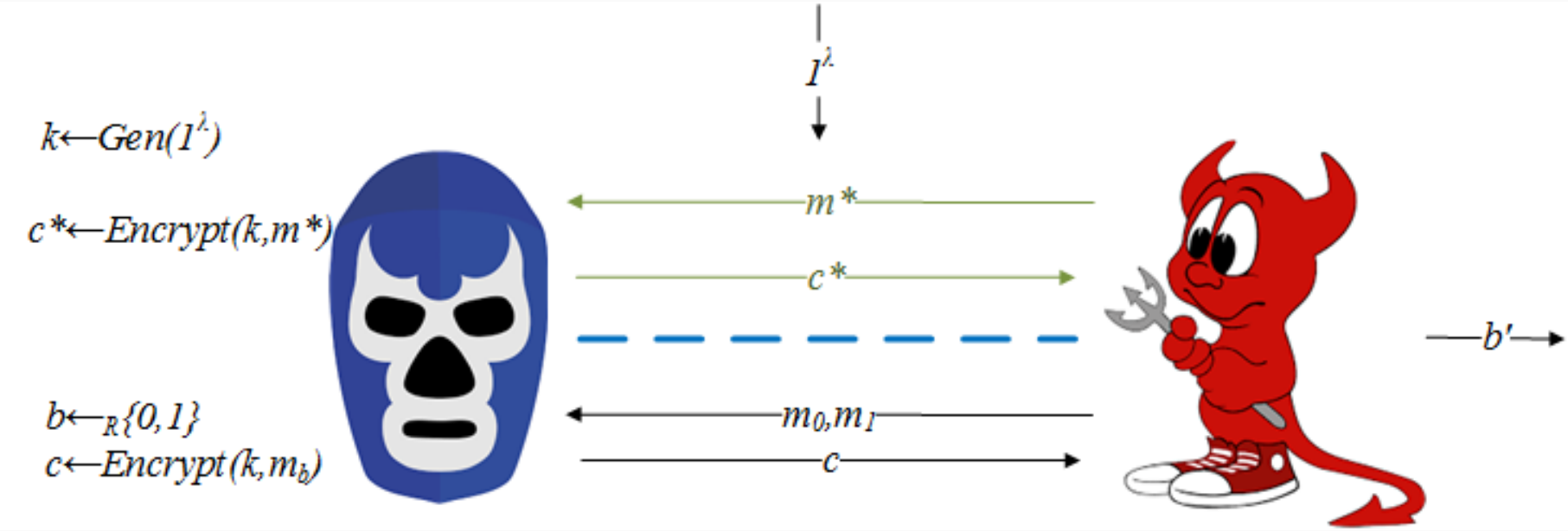
- $c_1^{t_1} c_2^{t_2} = m^{e_1 t_1} m^{e_2 t_2} = m^1 = m$

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

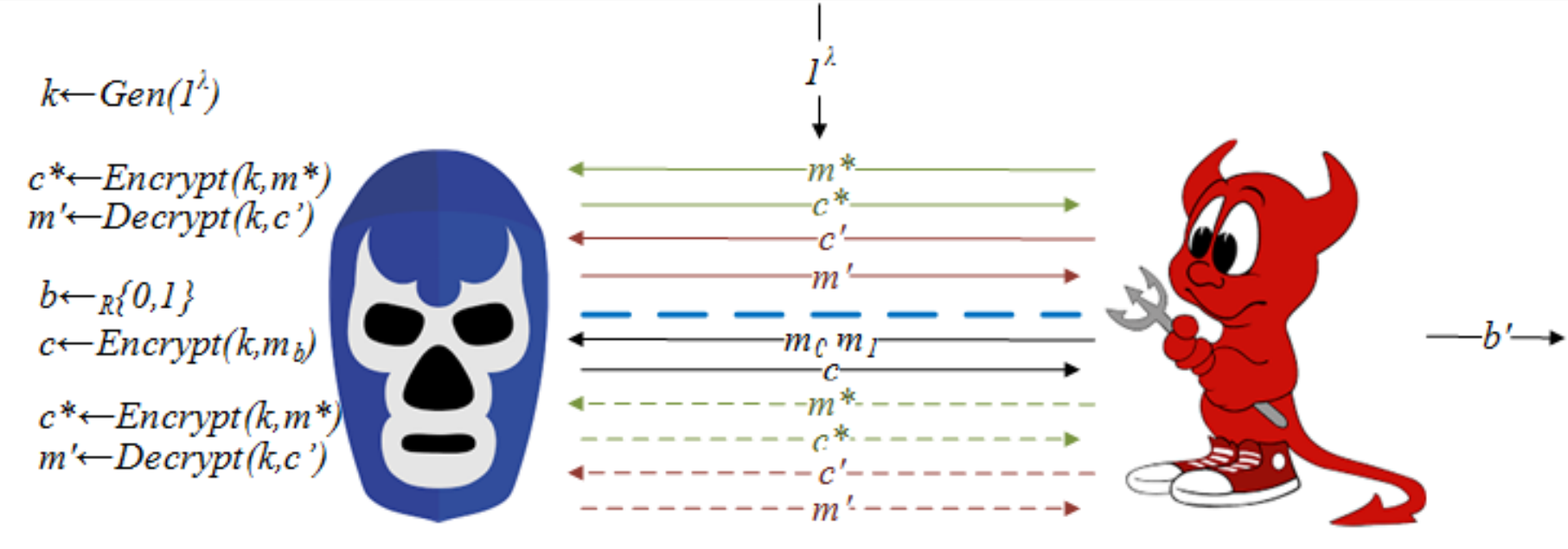
Μοντελοποίηση

Το (textbook) RSA δεν διαθέτει IND-CPA



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

Το RSA δεν διαθέτει IND-CCA



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

Το RSA είναι malleable

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

Στο παίγνιο CCA:

- Στόχος: Αποκρυπτογράφηση του $c_b = m_b^e \bmod n$
- Μπορεί να αποκρυπτογραφήσει το $c' = c_b x^e \bmod n$ 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 δεν διαρρέει
Έστω $c = m^e \bmod n$

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

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

Θεώρημα (Goldwasser, Micali, Tong)

Θεώρημα
Για κάθε στιγμιότυπο του RSA (e,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*

$loc \Rightarrow parity$

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

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

$loc(c) = 1 \Rightarrow m > \frac{n}{2} \Rightarrow 2m > n$ δηλ. $(2m \bmod n) \bmod 2 = 1$
αφού n μονός
και $2m \bmod n = 2m - n$ αφού $n < 2m < 2n$

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

$parity \Rightarrow loc$

Θα δείξουμε ότι: $parity(c) = loc(c \cdot \mathsf{Enc}(2^{-1}))$ Ισχύει:
$loc(c \cdot \mathsf{Enc}(2^{-1})) = loc(\mathsf{Enc}(m \cdot 2^{-1})) = loc(\mathsf{Enc}(m \cdot \frac{n+1}{2}))$

$parity(c) = 0 \Rightarrow m \bmod 2 = 0$ τότε: $\frac{m}{2} < \frac{n}{2}$ αφού $m < n$ και
$m \cdot \frac{n+1}{2} = \frac{m}{2} \pmod{n}$

$parity(c) = 1 \Rightarrow m \bmod 2 = 1$ τότε:

$(m\frac{n+1}{2}) \bmod n = ((2k+1)\frac{n+1}{2}) \bmod n =$
$[k(n+1) + \frac{n+1}{2}] \bmod n = k \bmod n + \frac{n+1}{2} > \frac{n}{2}$

Απόδειξη $(1) \Leftrightarrow (2) \Leftrightarrow (3)$

Προφανώς $(1) \Rightarrow (2)$ (αν μπορώ να αποκρυπτογραφήσω ξέρω parity) και $(2) \Leftrightarrow (3)$ (από προηγούμενα)

Για το $(3) \Rightarrow (1)$

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

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

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

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

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

... κ.ο.κ. για $\log n$ 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)

Η επίθεση του Bleichenbacher (Million Message Attack) i

Βασική Ιδέα:Padding Oracle
Χρήση ενός συστήματος το οποίο μπορεί να αποφανθεί αν ένα κρυπτοκείμενο έχει προκύψει με σωστό padding

- Ακριβής Μορφή padded μηνύματος στο pkcs1:
  $PKCS(r, m) = 0\text{x}\ \mathtt{00||02||r||00||m}$
- Μετά την αποκρυπτογράφηση:
  - Έλεγχος πρώτου byte για την τιμή 0
  - Έλεγχος δεύτερου byte για την τιμή 2
  - Αναζήτηση του 0
  - Ανάκτηση του m
- Το oracle στην πράξη:
  - Ύπαρξη μηνύματος λάθους για μη αποδεκτό padding ή
  - ανάκτηση της πληροφορίας μέσω side channel (πχ. χρόνος απάντησης)

Η επίθεση του Bleichenbacher (Million Message Attack) ii

Η επίθεση:

- Στόχος: Αποκρυπτογράφηση ενός c
- Ο $\mathcal{A}$ ξέρει ότι $c = PKCS(r, m)^e \bmod n$
- Επειδή είναι έγκυρο βρίσκεται σε ένα συγκεκριμένο εύρος τιμών (ξεκινούν με 0002)
  - $2 \cdot 2^{\lambda-16} \leq PKCS(r, m) \leq 3 \cdot 2^{\lambda-16} - 1$ όπου $\lambda = |n|$
- Διαλέγει *πολλά* τυχαία s
- Στέλνει στο padding oracle μηνύματα της μορφής $c' = s^e c \bmod n$
- Λόγω ιδιοτήτων RSA: $c' = (sPKCS(r, m))^e \bmod n$
- Στα περισσότερα η αποκρυπτογράφηση δίνει λάθος padding

Η επίθεση του Bleichenbacher (Million Message Attack) iii

- Αν *δεν* δώσει:
  - Ξέρουμε ότι το padded plaintext έχει σωστή μορφή δηλαδή: $c' = (PKCS(r', m'))^e \bmod n$
  - Αφού $PKCS(r', m') = sPKCS(r, m) \pmod n$ τότε:
    - $2 \cdot 2^{\lambda-16} \leq sPKCS(r, m) - kn \leq 3 \cdot 2^{\lambda-16} - 1$
    - Άρα $k = \frac{sPKCS(r,m) - PKCS(r',m')}{n}$
    - Το k μπορεί να περιοριστεί σε συγκεκριμένες τιμές από την γνώση για τα $sPKCS(r, m), PKCS(r', m')$
    - Για κάθε μία βρίσκουμε νέα διαστήματα για το $PKCS(r, m)$
  - Επαναλαμβάνουμε μέχρι να μείνει μία τιμή για το $PKCS(r, m)$
- Με 300.000 εως 2.000.000 c' μπορεί να αποκρυπτογραφηθεί το c

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

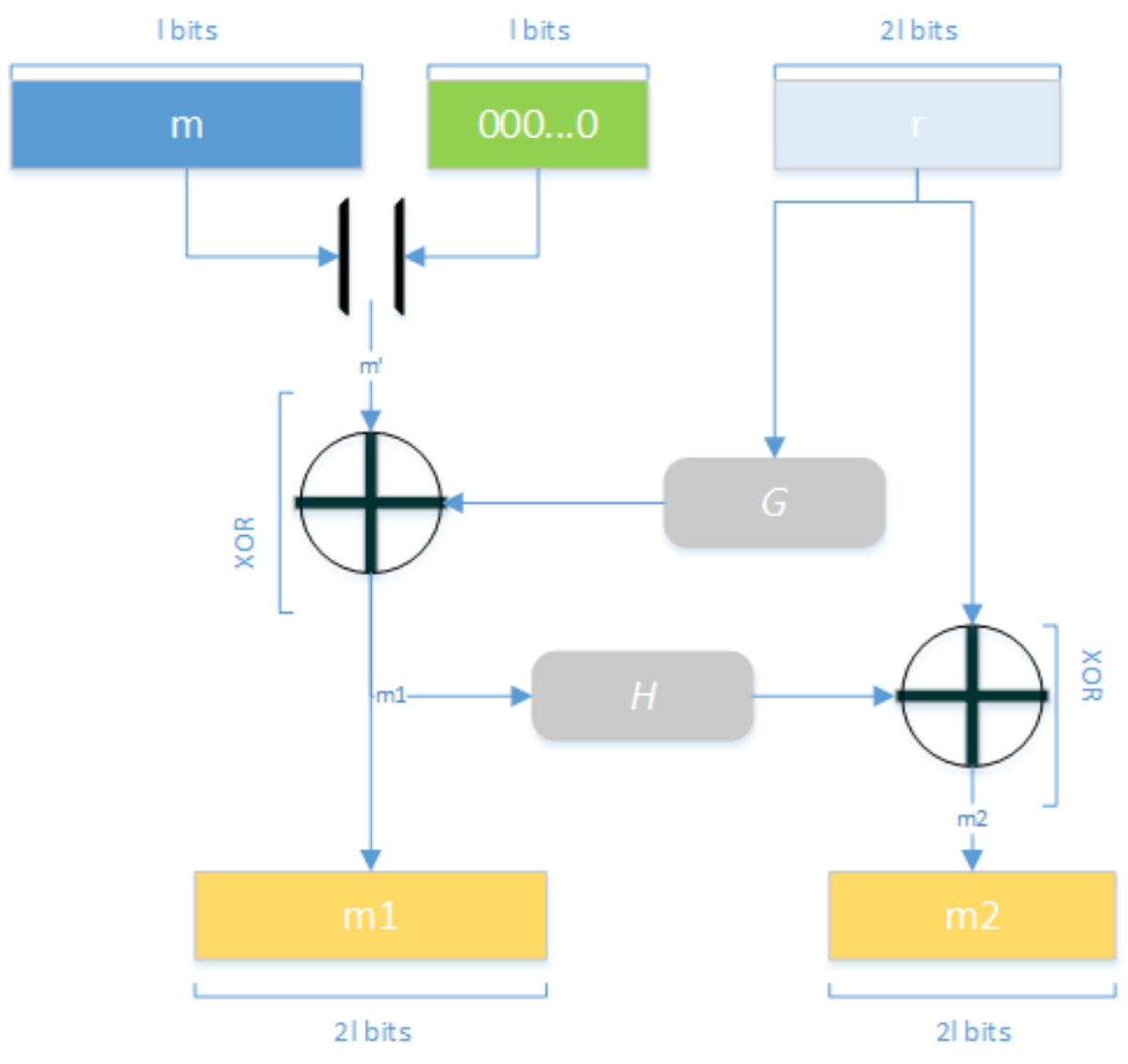
Τα τυχαία bits πρέπει να 'διαχυθούν' σε όλο το κρυπτοκείμενο (Δίκτυα Feistel)

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 $2l$: $m' = m||0^l$
- 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

Πηγές

Βιβλιογραφία

1. Παγουρτζής, Α., Ζάχος, Ε., ΓΠ, 2015. Υπολογιστική κρυπτογραφία. [ηλεκτρ. βιβλ.] Αθήνα:Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών
2. Jonathan Katz and Yehuda Lindell. Introduction to Modern Cryptography (Chapman and Hall/Crc Cryptography and Network Security Series). Chapman and Hall/CRC, 2007
3. Nigel Smart. Introduction to cryptography
4. Paar, Christof, and Jan Pelzl. Understanding cryptography: a textbook for students and practitioners. Springer Science-Business Media, 2009.
5. Dan Boneh, Introduction to cryptography, online course
6. R.L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21:120–126, 1978
7. Alexander May. Computing the rsa secret key is deterministic polynomial time equivalent to factoring. In Advances in Cryptology–CRYPTO 2004, pages 213–219. Springer, 2004.
8. Michael J Wiener. Cryptanalysis of short rsa secret exponents. Information Theory, IEEE Transactions on, 36(3):553–558, 1990.
9. Boneh, Dan. "Twenty years of attacks on the RSA cryptosystem." Notices of the AMS 46.2 (1999): 203-213.
10. Bellare, Mihir, and Phillip Rogaway. "Optimal asymmetric encryption." Advances in Cryptology—EUROCRYPT'94. Springer Berlin Heidelberg, 1995.
11. Bleichenbacher, Daniel. "Chosen ciphertext attacks against protocols based on the RSA encryption standard pkcs1" Advances in Cryptology—CRYPTO'98. Springer Berlin Heidelberg, 1998.
12. Lenstra, Arjen, James P. Hughes, Maxime Augier, Joppe Willem Bos, Thorsten Kleinjung, and Christophe Wachter. Ron was wrong, Whit is right.