Ψηφιακές Υπογραφές

Παναγιώτης Γροντάς - Άρης Παγουρτζής

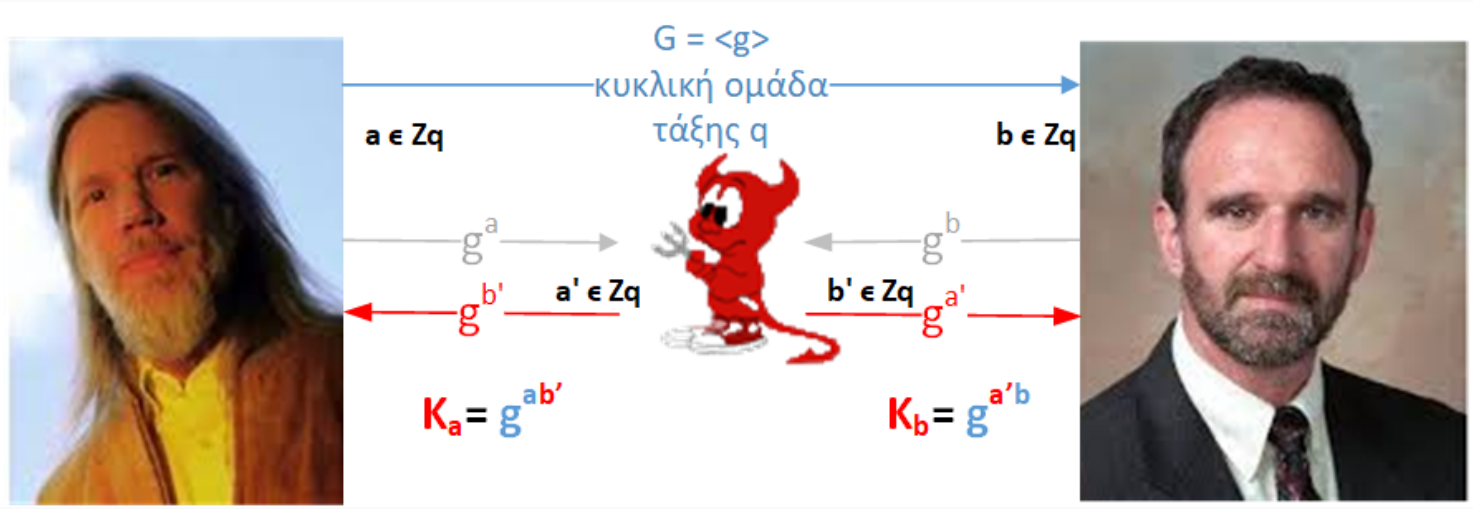
12/12/2023

ΕΜΠ - Κρυπτογραφία

Περιεχόμενα

Εισαγωγή

Εισαγωγή - Το πρόβλημα



Αποφυγή MITM attacks σε DHKE

- **Ακεραιότητα**: Το μήνυμα είναι *αυτό* που έστειλε ο αποστολέας
- **Αυθεντικοποίηση**: Το μήνυμα το έστειλε *αυτός* που φαίνεται ως αποστολέας

Μία λύση: MACs Μειονεκτήματα συμμετρικής κρυπτογραφίας

Ψηφιακές υπογραφές-Ασύμμετρα MACs

- Ο αποστολέας (υπογράφων S) εκτελεί αλγόριθμο KGen και παράγει τα $(\mathsf{sk}, \mathsf{vk})$
  - Το κλειδί επαλήθευσης πρέπει να είναι δημόσιο
  - Το κλειδί υπογραφής πρέπει να διατηρείται μυστικό
- Δημοσιοποιεί το κλειδί επαλήθευσης (web site, κατάλογο)
- Πριν την αποστολή μετασχηματίζει το μήνυμα (με το sk) παράγοντας την 'υπογραφή' σ
- Αποστέλλει το ζεύγος (m, σ)
  - Η υπογραφή *εξαρτάται* από το μήνυμα
  - Η υπογραφή είναι άχρηστη χωρίς το μήνυμα
- Ο παραλήπτης (επαληθεύων V) ελέγχει αν η υπογραφή που έλαβε είναι έγκυρη (με το vk)

Πλεονεκτήματα

- Εύκολη διανομή κλειδιού
- Δημόσια Επαληθευσιμότητα
  - Δεν επαληθεύει μόνο ο παραλήπτης
  - Δημόσιο κλειδί: Μπορεί να επαληθεύσει οποιοσδήποτε
- Μη αποκήρυξη (non repudiation)
  - Αντιμετώπιση εσωτερικού αντίπαλου που προσπαθεί να αρνηθεί τις υπογραφές του
  - Μαθηματική σχέση κλειδιών υπογραφής - επαλήθευσης
- Αυθεντικοποίηση χρηστών
  - Λόγω *κατοχής* του ιδιωτικού κλειδιού
- Επιπλέον λειτουργίες
  - Ανωνυμία (*τυφλές* υπογραφές)
  - Αντιπροσώπευση από ομάδα (με σταθερά μέλη: *ομαδικές* υπογραφές, χωρίς: υπογραφές *δακτυλίου*)
  - Με προκαθορισμένο επαληθευτή

Μειονεκτήματα

Λύσαμε τα προβλήματα **διανομής** κλειδιού, **αυθεντικότητας** και **ακεραιότητας** μηνύματος

Δημιουργήσαμε το πρόβλημα **αυθεντικότητας** κλειδιού

- Πώς είμαστε σίγουροι πως το ζεύγος κλειδιών αντιστοιχεί όντως στον S;
- Πώς είμαστε σίγουροι πώς το sk ήταν στην κατοχή του S κατά τη δημιουργία της υπογραφής;

Μαθηματικές και μη λύσεις

Ορισμός

Σχήμα Υπογραφής
Μια τριάδα από αλγόριθμους

- $\mathsf{KGen}(1^\lambda) = (\mathsf{sk}, \mathsf{vk})$
- $\mathsf{Sign}(\mathsf{sk}, m) = \sigma, \qquad m \in \{0,1\}^*$
- $\mathsf{Vf}(\mathsf{vk}, m, \sigma) \in \{0,1\}$

Έγκυρες υπογραφές: επαληθεύονται με επιτυχία.

Ορθότητα
$\mathsf{Vf}(\mathsf{vk}, m, \mathsf{Sign}(\mathsf{sk}, m)) = 1 \;\; \forall m, (\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KGen}$

Επιθέσεις

Πλαστογραφία(Forgery)
Ο $\mathcal{A}$ με δεδομένα το δημόσιο κλειδί επαλήθευσης και ένα μήνυμα παράγει μια έγκυρη υπογραφή χωρίς την συμμετοχή του S.

Στόχοι αντιπάλου

- Καθολική πλαστογράφηση: Ο $\mathcal{A}$ μπορεί να παράγει έγκυρες υπογραφές σε όποιο μήνυμα θέλει ($\Leftrightarrow$ κατοχή ιδιωτικού κλειδιού)
- Επιλεκτική πλαστογράφηση: Ο $\mathcal{A}$ μπορεί να παράγει 1 έγκυρη υπογραφή σε μήνυμα (με νόημα) της επιλογής του
- Υπαρξιακή πλαστογράφηση: Ο $\mathcal{A}$ μπορεί να παράγει 1 έγκυρη υπογραφή (τυχαία bits) σε τυχαίο μήνυμα

Είδη Αντιπάλων - Επιθέσεων

- Παθητικός (passive): Απλά γνωρίζει το κλειδί επαλήθευσης και ζεύγη μηνυμάτων, έγκυρων υπογραφών
- Ενεργός (active): Μπορεί να αποκτήσει έγκυρες υπογραφές σε μηνύματα της επιλογής του (chosen message attack)
- Ενεργός με προσαρμοστικότητα (adaptive active): Μπορεί να αποκτήσει έγκυρες υπογραφές σε μηνύματα της επιλογής του που εξαρτώνται από προηγούμενες έγκυρες υπογραφές

Αντίπαλοι ii

Ασφάλεια ως προς τον δυνατότερο αντίπαλο - γενικότερη επίθεση

Ασφάλεια
Ένα σχήμα υπογραφής είναι ασφαλές αν δεν επιτρέπει σε έναν ενεργό αντίπαλο με προσαρμοστικότητα να επιτύχει υπαρξιακή πλαστογράφηση σε επίθεση επιλεγμένων μηνυμάτων (EUF-CMA).

Ορισμός Ασφάλειας

Το παιχνίδι πλαστογράφησης Forge-Game

- Ο S εκτελεί τον αλγόριθμο $\mathsf{KGen}(1^\lambda)$ και παράγει τα $(\mathsf{vk}, \mathsf{sk})$
- Ο $\mathcal{A}$ έχει πρόσβαση σε ένα μαντείο υπογραφών $\mathcal{A}^{\mathsf{Sign}}(\cdot)$ με το οποίο αποκτά ένα σύνολο έγκυρων υπογραφών $Q = \{(m_i, \sigma_i)\}$ σε μηνύματα της επιλογής του
- Ο $\mathcal{A}$ επιλέγει ένα μήνυμα m και παράγει το ζεύγος (m, σ)
- Νίκη $\mathcal{A} : \mathsf{Forge-Game}(\mathcal{A}) = 1 \Leftrightarrow \mathsf{Vf}(\mathsf{vk}, m, \sigma) = 1 \wedge m \notin Q$

Ο $\mathcal{A}$ κερδίζει το παιχνίδι αν $\Pr[\mathsf{Forge-Game}(\mathcal{A}) = 1] \geq negl(\lambda)$

Παραλλαγή: **strong** $\mathrm{EUF\text{-}CMA}$:
Νίκη $\mathcal{A} : \mathsf{Forge-Game}(\mathcal{A}) = 1 \Leftrightarrow \mathsf{Vf}(\mathsf{vk}, m, \sigma) = 1 \wedge (m, \sigma) \notin Q$

Ψηφιακές Υπογραφές RSA

Ψηφιακές Υπογραφές RSA

Δημιουργία Κλειδιών: $\mathsf{KGen}(1^\lambda) = (d, (e, n))$

- $n = p \cdot q$, p, q πρώτοι αριθμοί $\frac{\lambda}{2}$ bits
- Επιλογή e ώστε $gcd(e, \phi(n)) = 1$
- $d = e^{-1} \pmod{\phi(n)}$ με EGCD

Υπογραφή - 'Αποκρυπτογράφηση' RSA

- $\mathsf{Sign}(d, m) = m^d \bmod n$

Επαλήθευση - 'Κρυπτογράφηση' RSA

- $\mathsf{Vf}((e, n), m, \sigma) = \quad \sigma^e =_? m \pmod{n}$

Ορθότητα
$\mathsf{Vf}((e, n), m, m^d) = \quad (m^d)^e = m \pmod{n}$

...αλλά καθόλου ασφάλεια

Επίθεση Χωρίς Μήνυμα (No message attack)

- Ο $\mathcal{A}$ έχει στη διάθεση του δημόσιο κλειδί (e, n)
- $Q = \emptyset$ - δεν υποβάλλονται μηνύματα για υπογραφή
- Επιλογή τυχαίου $\sigma \in \mathbb{Z}_n^*$
- 'Κρυπτογράφηση' σ: $\sigma^e \bmod n$
- Παράγεται κάποιο στοιχείο $\in \mathbb{Z}_n*$. Το βαφτίζουμε m
- Το ζεύγος (m, σ) είναι έγκυρο και $(m, \sigma) \notin Q$
- Ο $\mathcal{A}$ κερδίζει με πιθανότητα 1

Έχει νόημα; - Ναι, μπορεί m να είναι κάποια δυαδική κωδικοποίηση. Με επαναλήψεις μπορούν να βρεθούν m όπου κάποια bits μπορεί να αντιστοιχούν σε έγκυρα τμήματα μηνυμάτων.

Επίθεση Επιλεγμένων Μηνυμάτων (Chosen message attack)

- Ο $\mathcal{A}$ έχει στη διάθεση του δημόσιο κλειδί (e, n) και θέλει να πλαστογραφήσει υπογραφή για κάποιο $m \in \mathbb{Z}_n^*$
- Ο $\mathcal{A}$ χρησιμοποιώντας το μαντείο **αποκτά τις υπογραφές** 2 μηνυμάτων $Q = \{(m_1, \sigma_1), (\frac{m}{m_1}, \sigma_2)\}$ με $m_1 \xleftarrow{\$} \mathbb{Z}_n^*$
- Υπολογισμός $\sigma = \sigma_1\sigma_2 = m_1^d(\frac{m}{m_1})^d = m^d \pmod n$
- Η σ είναι έγκυρη υπογραφή για το m και $\notin Q$

Blinding - Επίθεση Επιλεγμένων Μηνυμάτων (Chosen message attack)

- Ο $\mathcal{A}$ έχει στη διάθεση του δημόσιο κλειδί (e, n) και θέλει να πλαστογραφήσει υπογραφή σ για κάποιο $m \in \mathbb{Z}_n^*$
- Ο $\mathcal{A}$ επιλέγει $r \leftarrow\$ \mathbb{Z}_n^*$ και υπολογίζει $m' := m \cdot r^e \bmod n$
- Χρησιμοποιεί το μαντείο και αποκτά την υπογραφή $\sigma' = m'^d = m^d \cdot r^{ed} = m^d r = \sigma r \pmod n$
- Ανάκτηση $\sigma := \sigma' \cdot r^{-1} \bmod n$
- Η σ είναι έγκυρη υπογραφή για το m και $\notin Q$

Λύση: Υπόδειγμα Hash - and - Sign.

RSA - FDH (Full Domain Hash) i

Δημιουργία Κλειδιών: $\mathsf{KGen}(1^\lambda) = (d, (e, n))$

- $n = p \cdot q$, p, q πρώτοι αριθμοί $\frac{\lambda}{2}$ bits
- Επιλογή e ώστε $gcd(e, \phi(n)) = 1$
- $d = e^{-1} \pmod{\phi(n)}$ με EGCD
- Χρήση δημόσια διαθέσιμης τυχαίας συνάρτησης $\mathsf{H} : \{0,1\}^* \to \mathbb{Z}_n^*$

Πρακτικά: $\mathsf{FDH}(m) = \mathsf{H}(m||0)||\mathsf{H}(m||1)\cdots$ (PKCS 1)

ή μπορεί να υποστηρίζεται natively από την H

RSA - FDH (Full Domain Hash) ii

Υπογραφή

- Υπολογισμός $\mathsf{H}(m)$
- $\mathsf{Sign}(d, m) = \mathsf{H}(m)^d \bmod n$

Επαλήθευση

- Υπολογισμός $\mathsf{H}(m)$
- $\mathsf{Vf}((e, n), m, \sigma) = \quad \sigma^e =_? \mathsf{H}(m) \pmod{n}$

RSA - FDH (Full Domain Hash) iii

Ορθότητα

$$\mathsf{Vf}((e,n),m,\mathsf{H}(m)^d) = \quad (\mathsf{H}(m)^d)^e = \mathsf{H}(m) \pmod{n}$$

Υλοποίηση: συνάρτηση σύνοψης με δυσκολία εύρεσης συγκρούσεων

Πλεονέκτημα: Μπορεί να χρησιμοποιηθεί για υπογραφή τυχαίων συμβολοσειρών και όχι μόνο στοιχείων του $\mathbb{Z}_n^*$

RSA - FDH (Full Domain Hash) iv

- Επίθεση χωρίς μήνυμα
  - Επιλογή τυχαίου $\sigma \in \mathbb{Z}_n^*$
  - Η ‘κρυπτογράφηση‘ δίνει τη σύνοψη $h = \sigma^e \pmod{n}$ - όχι το μήνυμα
  - Για το μήνυμα πρέπει να βρεθεί $m : \mathrm{H}(m) = h$
  - Δυσκολία αντιστροφής

RSA - FDH (Full Domain Hash) v

- Επίθεση επιλεγμένων μηνυμάτων
  - Ο $\mathcal{A}$ έχει στη διάθεση του δημόσιο κλειδί (e, n) και θέλει να πλαστογραφήσει υπογραφή για $m \in \mathbb{Z}_n^*$
  - Ο $\mathcal{A}$ χρησιμοποιώντας το μαντείο αποκτά τις υπογραφές 2 μηνυμάτων $Q = \{(m_1, \sigma_1), (\frac{m}{m_1}, \sigma_2)\}$ με $m_1 \leftarrow\$ \mathbb{Z}_n^*$
  - Υπολογισμός $\sigma := \sigma_1\sigma_2 = \mathrm{H}(m_1)\mathrm{H}(\frac{m}{m_1})$
  - Δεν ισχύουν οι ομομορφικές ιδιότητες.

- Απόδειξη Ασφάλειας: Πρέπει η H να δίνει 'τυχαίες' τιμές
- Αρκούν οι ιδιότητες τους (one-way-ness, collision resistance);
  - ΟΧΙ
- Το μοντέλο του τυχαίου μαντείου (M. Bellare, P. Rogaway, -1993)

Το μοντέλο του τυχαίου μαντείου

Συναρτήσεις σύνοψης ως τυχαίες συναρτήσεις - informal

- Θεωρητικά θα θέλαμε να συμπεριφέρονται ως τυχαίες συναρτήσεις
- Πρακτικά όμως: αδύνατον να κατασκευαστούν
  - Συνάρτηση $H : \{0,1\}^n \to \{0,1\}^{l(n)}$
  - Κατασκευή ως πίνακας τιμών: Απαιτούνται 2^n γραμμές

Έισοδος	Έξοδος
$0 \cdots 00$	r_1
$0 \cdots 01$	r_2
$\cdots$	$\cdots$
$1 \cdots 11$	$r_{l(n)}$

  - Συμπίεση: Μείωση τυχαιότητας

Ακόμα και να μπορούσαν να κατασκευαστούν
αδύνατη αποθήκευση
εκθετική αποτίμηση (μη αποδεκτή και για χρήστη και για αντίπαλο)
Επίσης συνήθως $H : \{0,1\}^* \to \{0,1\}^{l(n)}$

Συναρτήσεις σύνοψης και αποδείξεις ασφάλειας i

Τυχαίο Μαντείο - Αφαιρετική αναπαράσταση τέλειας συνάρτησης σύνοψης

- Μαύρο κουτί - απαντάει σε ερωτήσεις
- (Τέλεια) Ασφάλεια στο κανάλι επικοινωνίας (μοντελοποίηση τοπικής αποτίμησης)
- Είναι συνάρτηση (ίδια είσοδος - ίδια έξοδος σε κάθε κλήση)
- Είναι συνάρτηση σύνοψης (υπάρχουν συγκρούσεις - αλλά είναι δύσκολο να βρεθούν)

Συναρτήσεις σύνοψης και αποδείξεις ασφάλειας ii

Δυναμική κατασκευή - Lazy Evaluation

- Εσωτερικός πίνακας - αρχικά άδειος
- Για κάθε ερώτηση: έλεγχος αν έχει ήδη απαντηθεί
- Αν ναι, τότε ανάκτηση της απάντησης
- Αν όχι, απάντηση με τυχαία τιμή και αποθήκευση για μελλοντική αναφορά

Συναρτήσεις σύνοψης και αποδείξεις ασφάλειας iii

Αποδείξεις στο μοντέλο τυχαίου μαντείου (Bellare - Rogaway)

- Ο $\mathcal{A}$ *νομίζει* ότι αλληλεπιδρά με το τυχαίο μαντείο
- Στην πραγματικότητα το προσομοιώνει η αναγωγή (programmability)
- Μπορούμε να μάθουμε τις ερωτήσεις του $\mathcal{A}$
- Στο πραγματικό πρωτόκολλο το τυχαίο μαντείο αντικαθίσταται από μία πραγματική συνάρτηση (πχ. SHA256)

Απόδειξη Ασφάλειας RSA-FDH

Θεώρημα
Αν το πρόβλημα RSA είναι δύσκολο, τότε οι υπογραφές RSA-FDH παρέχουν ασφάλεια έναντι υπαρξιακής πλαστογράφησης με επιλεγμένα μηνύματα $\mathrm{EUF\text{-}CMA}$ στο μοντέλο του τυχαίου μαντείου.

Θα αποδείξουμε το παρακάτω:

Θεώρημα
Αν υπάρχει αντίπαλος $\mathcal{F}$ που παράγει πλαστογράφηση στο RSA-FDH με πιθανότητα τουλάχιστον p_F μετά από q_H ερωτήσεις στο τυχαίο μαντείο, τότε υπάρχει αντίπαλος $\mathcal{R}$ που λύνει το πρόβλημα RSA με πιθανότητα $r \geq \frac{p_F}{q_H}$.

Απόδειξη Ασφάλειας Hashed RSA: Κατασκευή $\mathcal{R}$

- Ο $\mathcal{F}$ μπορεί να κατασκευάσει πλαστογράφηση υπογραφής
- Κατασκευή $\mathcal{R}$ που με χρήση του $\mathcal{F}$ και ενός τυχαίου μαντείου μπορεί να αντιστρέψει το RSA
- Είσοδος $\mathcal{R}$
  - Δημόσιο κλειδί (e, n)
  - Ομοιόμορφα επιλεγμένο $y \leftarrow\!\!\$\ \mathbb{Z}_n^*$
- Έξοδος $\mathcal{R}$
  - $x = y^{e^{-1}}$

Απόδειξη Ασφάλειας Hashed RSA
Επίθεση χωρίς μήνυμα i

Υπόθεση
Για την πλαστογράφηση (m^*, σ^*) έχει προηγουμένως ερωτηθεί στο μαντείο το $\mathsf{H}(m^*)$

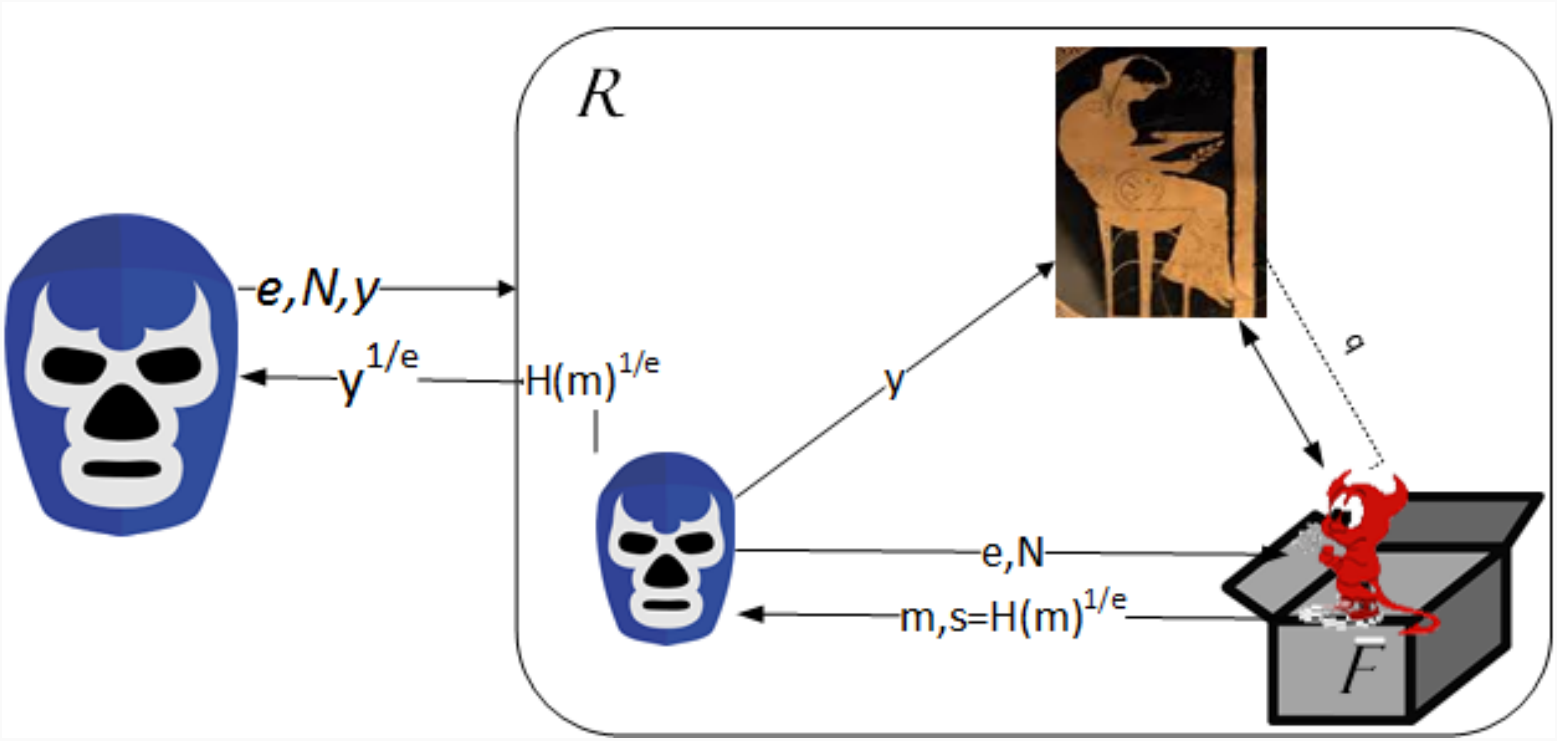
Συνέπεια
Εφόσον η πλαστογράφηση είναι έγκυρη υπογραφή πρέπει $\sigma^{*e} = \mathsf{H}(m^*)$ Άρα $\sigma^* = \mathsf{H}(m^*)^{e^{-1}}$

Πλήθος ερωτήσεων
Ο $\mathcal{R}$ δεν γνωρίζει ακριβώς το q_{H}, αλλά δεν έχει σημασία αφού $\mathcal{F}$ είναι PPT. Άρα $q_{\mathsf{H}} \in \mathcal{O}(()\mathrm{poly}(\lambda))$.

Απόδειξη Ασφάλειας Hashed RSA
Επίθεση χωρίς μήνυμα ii

- Ο $\mathcal{R}$ προωθεί το (e, n) στον $\mathcal{F}$
- Ο $\mathcal{R}$ επιλέγει $j \leftarrow\$ [q_{\mathsf{H}}]$
- Ο $\mathcal{F}$ όταν ρωτάει το μαντείο H για μηνύματα $\{m_i\}_{i=1}^{q_{\mathsf{H}}}$ λαμβάνει τις απαντήσεις $\{\mathsf{H}(m_i)\}_{i=1}^{q_{\mathsf{H}}} \in_r \mathbb{Z}_n^*$
- Ο $\mathcal{R}$ *ελπίζει* ότι στο ερώτημα j θα γίνει η πλαστογράφηση
- Αν $i = j$, ο $\mathcal{R}$ αντικαθιστά την απάντηση $\mathsf{H}(m_j^*)$ με το y
- Αν έχει δίκιο, τότε ο $\mathcal{F}$ εξάγει την πλαστογραφία (m_j^*, σ^*) με πιθανότητα p_F
- Δηλαδή: $\sigma^{*e} = y \Rightarrow \sigma^* = y^{e^{-1}}$
- Ο $\mathcal{R}$ προωθεί το σ^* στην έξοδο
- Ο $\mathcal{R}$ αντέστρεψε το RSA με πιθανότητα επιτυχίας $\frac{p_F}{q_{\mathsf{H}}}$

Απόδειξη Ασφάλειας Hashed RSA
Επίθεση χωρίς μήνυμα iii



Απόδειξη Ασφάλειας Hashed RSA
Επίθεση επιλεγμένου μηνύματος i

Σενάριο

- $\mathcal{F}$ ζητάει συνόψεις και υπογραφές από τον $\mathcal{R}$
- Συνόψεις: το τυχαίο μαντείο - προσομοίωση
- Υπογραφές: Πρέπει να τις απαντήσει ο $\mathcal{R}$
- δηλ. να υπολογίσει το $H(m)^d$ χωρίς το ιδιωτικό κλειδί...

Απόδειξη Ασφάλειας Hashed RSA
Επίθεση επιλεγμένου μηνύματος ii

Λύση

Ερώτηση $\mathcal{F}^{\mathsf{H}}(m)$:

Επιλογή $\sigma \leftarrow\!\!\$\ \mathbb{Z}_n^*$,

υπολογισμός σ^e και επιστροφή αντί $\mathsf{H}(m)$,

Αποθήκευση σ, σ^e, m για μετά

Ερώτηση $\mathcal{F}^{\mathcal{S}}(m)$:

Επιστροφή σ από την αντίστοιχη απάντηση για $\mathsf{H}(m)$.

Τετριμμένη επαλήθευση $\sigma^e = \mathsf{H}(m) = \sigma^e$

Απόδειξη Ασφάλειας Hashed RSA
Επίθεση επιλεγμένου μηνύματος iii

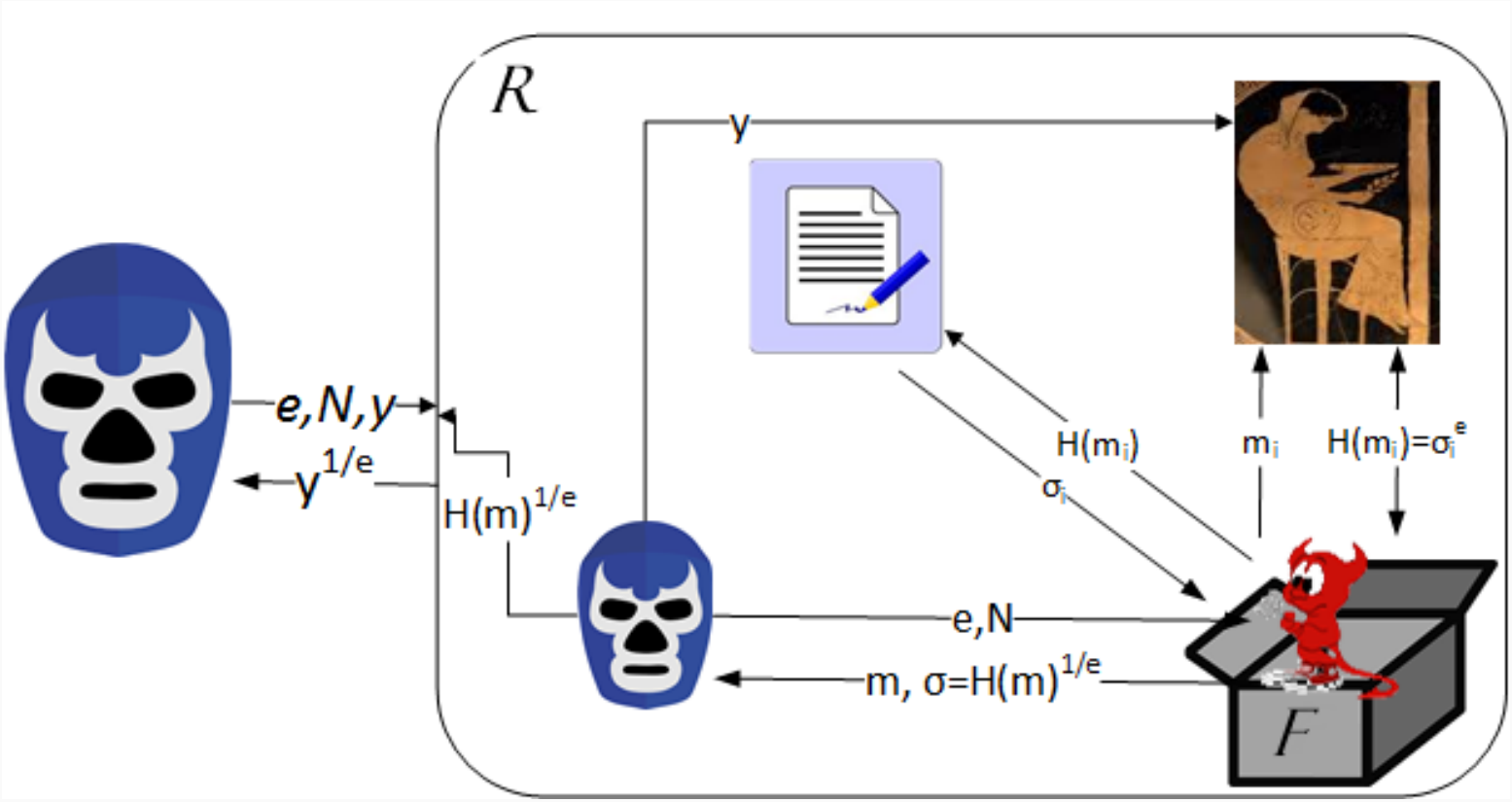
- Ο $\mathcal{R}$ προωθεί το (e, n) στον $\mathcal{F}$
- Ο $\mathcal{R}$ επιλέγει $j \leftarrow\$ [q_{\mathsf{H}}]$
- Ο $\mathcal{F}$ κάνει $q_{\mathsf{H}} = O(\text{poly}(\lambda))$ ερωτήσεις στο μαντείο για μηνύματα $\{m_i\}_{i=1}^{q_{\mathsf{H}}}$
- Κάθε ερώτηση του μαντείου H απαντάται από τον $\mathcal{R}$ ως εξής:
  - Επιλέγει τυχαίο $\sigma_i \in \mathbb{Z}_n^*$
  - Υπολογίζει $\sigma_i^e = y_i$ και θέτει $\mathsf{H}(m_i) = y_i$
  - Επιστρέφει y_i
  - Αποθηκεύει τις τριάδες $\mathcal{T} = (y_i, \sigma_i, m_i)$
- Ο $\mathcal{F}$ ζητάει υπογραφές από το μαντείο υπογραφών
  - Για κάθε σύνοψη y_i γίνεται αναζήτηση στον $\mathcal{T}$ για την τριάδα που περιέχει το y και επιστρέφεται το σ_i

Απόδειξη Ασφάλειας Hashed RSA
Επίθεση επιλεγμένου μηνύματος iv

- Οι υπογραφές είναι έγκυρες αφού $\sigma_i^e = y_i$

- Ο $\mathcal{R}$ απαντάει το ερώτημα j στο H με y
- Για το συγκεκριμένο δεν θα ζητηθεί υπογραφή, αλλά το σ^* θα παραχθεί από τον $\mathcal{F}$ (πλαστογράφηση)
  - Αν ζητηθεί abort
- Αφού πλαστογράφηση = έγκυρη υπογραφή θα ισχύει $\sigma^{*e} = y$, δηλαδή $\sigma^* = y^{e^{-1}}$
- Άρα ο $\mathcal{R}$ πέτυχε το στόχο του και αντέστρεψε το y
- Πιθανότητα επιτυχίας $\mathcal{F}$ p_F και πιθανότητα επιτυχίας $\mathcal{R}$: $\frac{p_F}{q_H}$

Απόδειξη Ασφάλειας Hashed RSA
Επίθεση επιλεγμένου μηνύματος v



Απόδειξη Ασφάλειας Hashed RSA
Επίθεση επιλεγμένου μηνύματος vi

Αν το πρόβλημα RSA είναι δύσκολο:

$r \leq \text{negl}(\lambda)$ τότε
$p_F \leq r \cdot q_H \leq \text{negl}(\lambda)$.

Παρατήρηση
Τα παραπάνω ισχύουν ασυμπτωτικά.

Πρακτικά υπάρχει 'απώλεια' ασφάλειας q_H.

Π.χ.: Αν $r = 2^{-60}$ και $q_H = 2^{50}$ τότε $p_F = 2^{-10}$.

Το μοντέλο του τυχαίου μαντείου - κριτική

Μειονεκτήματα

’Άχρηστη’ απόδειξη - Καμία πραγματική συνάρτηση H δεν είναι random oracle

Programmability - Η περιγραφή της συνάρτησης είναι σταθερή στην πραγματικότητα

Ύπαρξη ’θεωρητικών’ σχημάτων τα οποία αποδεικνύονται ασφαλή, αλλά οποιαδήποτε κατασκευή τους είναι μη ασφαλής

Πλεονεκτήματα

Η απόδειξη εστιάζει στο πρωτόκολλο και όχι στο H

Απόδειξη με χρήση τυχαίου μαντείου είναι καλύτερη από απουσία απόδειξης

Η μόνη αδυναμία: η συνάρτηση σύνοψης

Δεν υπάρχουν πραγματικές επιθέσεις που να έχουν εκμεταλλευτεί την απόδειξη μέσω τυχαίου μαντείου

Ψηφιακές Υπογραφές Διακριτού Λογάριθμου

Σχήμα Υπογραφής ElGamal i

Taher ElGamal (1985)

Δημιουργία Κλειδιών (KGen(1^λ))**:**

- Επιλογή πρώτου p. Δουλεύουμε στο $\mathbb{Z}_p^*$!
- Επιλογή γεννήτορα g
- Επιλογή $x \leftarrow\!\!\$\ \{1, \cdots, p-2\}$
- Υπολογισμός του $y = g^x \bmod p$
- Δημόσιο κλειδί (p, g, y), ιδιωτικό κλειδί x.

Σχήμα Υπογραφής ElGamal ii

Υπογραφή Μηνύματος m

- Επιλογή εφήμερου κλειδιού (**nonce**) $k \leftarrow\!\!\$\ \mathbb{Z}_{p-1}^*$ με $gcd(k, p-1) = 1$
- Υπολογισμός

$$r := g^k \bmod p$$
$$s := (m - xr)k^{-1} \bmod (p-1)$$

- Αν $s = 0$ επανάληψη με νέο k
- Υπογραφή είναι: $\sigma = (r, s)$
- Δύο αριθμοί μεγέθους $|p|$ bits

Σχήμα Υπογραφής ElGamal iii

Επαλήθευση υπογραφής

$$\mathrm{Vf}(y, m, (r, s)) = \begin{cases} 1, & y^r \cdot r^s = g^m \pmod p \\ 0, & y^r \cdot r^s \neq g^m \pmod p \end{cases}$$

Ορθότητα

Λόγω της κατασκευής του s

$$y^r r^s = g^{xr} g^{ks} = g^{xr+ks} = g^m \pmod p$$

Παρατηρήσεις

- Πιθανοτικό σχήμα υπογραφής - πολλές έγκυρες υπογραφές για ένα μήνυμα m (διαφορετικά k)
- Η συνάρτηση επαλήθευσης δέχεται οποιαδήποτε από αυτές ως έγκυρη
- Χειρισμός Τυχαιότητας
  - Το τυχαία επιλεγμένο k πρέπει να κρατείται κρυφό Αλλιώς ανάκτηση x από $s = (m - xr)k^{-1} \bmod (p-1)$
  - Η επανάληψη της χρήσης του ίδιου k καθιστά εφικτό τον υπολογισμό του

Επίθεση επανάληψης κλειδιού

Χρήση **ίδιου** εφήμερου κλειδιού k στην υπογραφή μηνυμάτων m_1, m_2

- $\text{Sign}(x, m_1) = (r, s_1)$ με $s_1 = (m_1 - xr)k^{-1}$
- $\text{Sign}(x, m_2) = (r, s_2)$ με $s_2 = (m_2 - xr)k^{-1}$
- Υπολογισμός

  $s_1 - s_2 = (m_1 - m_2)k^{-1} \Rightarrow (s_1 - s_2)k = (m_1 - m_2) \pmod{p-1}$
  - Αν $gcd(s_1 - s_2, p - 1) = 1$: $k = (m_1 - m_2)(s_1 - s_2)^{-1}$
  - Αλλιώς:

    Εύρεση k με δοκιμές από τις $gcd(s_1 - s_2, p - 1)$ πιθανές λύσεις και επαλήθευση με $r = g^k$ (αν είναι μικρός ο αριθμός)

Επίθεση επανάληψης κλειδιού (2)

- Έστω $d = gcd(s_1 - s_2, p - 1)$
- $d|(p-1)$ και $d|(s_1 - s_2)$. Άρα $d|(m_1 - m_2)$
- Θέτουμε $m' = \frac{m_1 - m_2}{d}, s' = \frac{s_1 - s_2}{d}, p' = \frac{p-1}{d}$
- Άρα: $s'k = m' \pmod{p'}$
- και $k = m'(s')^{-1} \pmod{p'}$ αφού $gcd(s', p') = 1$
- d λύσεις: $k = m'(s')^{-1} + ip' \bmod p - 1$ με $i \in \{0, \cdots, d-1\}$
- Δοκιμάζουμε ποια από αυτές επαληθεύει την $r = g^k \pmod{p}$

Ασφάλεια έναντι πλαστογράφησης i

Στόχος: Ευρεση m ώστε $y^r \cdot r^s = g^m \pmod p$

1. No message attack:
   Επιλέγω r και s, ψάχνω m: Επίλυση DLP.
2. Chosen message attack:
   Επιλέγω m και προσπαθώ να βρω r, s:
   - Επιλέγω r, ψάχνω s: $r^s = g^m \cdot y^{-r} \pmod p$ (επίλυση DLP).
   - Επιλέγω s, ψάχνω r: $y^r = g^m \cdot r^{-s} \pmod p$
     Ανοιχτό πρόβλημα - δε γνωρίζουμε σχέση με DLP

Ασφάλεια έναντι πλαστογράφησης ii

3. Κατασκευή r, s, m ταυτόχρονα.
   Επιλέγω i, j με $0 \leq i, j \leq p-2$, και $\gcd(j, p-1) = 1$:

$$r := g^i \cdot y^j \bmod p$$
$$s := -r \cdot j^{-1} \bmod (p-1)$$
$$m := -r \cdot i \cdot j^{-1} \bmod (p-1)$$

Τα $(m, (r, s))$ αποτελούν έγκυρο ζεύγος μηνύματος - υπογραφής

$$y^r \cdot r^s = y^r \cdot (g^{is} \cdot y^{js}) = g^{is} y^{r+js} = g^{-rij^{-1}} = g^m$$

Εφικτό σενάριο, δίνει υπογραφή για τυχαίο m (υπαρξιακή πλαστογράφηση)
Αντιμετώπιση με συνάρτηση σύνοψης

Υπογραφές Schnorr

- $\mathsf{KGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk}) = (g^x, x)$
- Υπογραφή:
  - Τυχαία επιλογή $t \leftarrow\!\!\$\ \mathbb{Z}_q$,
  - Υπολογισμός $T := g^t \bmod p$
  - Υπολογισμός $c := \mathsf{H}(T||\mathsf{pk}||m)$ όπου H είναι μια συνάρτηση σύνοψης που δίνει τιμές στο $\mathbb{Z}_q$
  - Υπολογισμός $s = t + cx \bmod q$ (για να μην χρειαστεί αντίστροφος μετά)
  - Δημοσιοποίηση του $\sigma := (T, s)$
- Επαλήθευση (από οποιονδήποτε)

$$c := \mathsf{H}(T||\mathsf{pk}||m)$$
$$g^s = T\mathsf{pk}^c$$

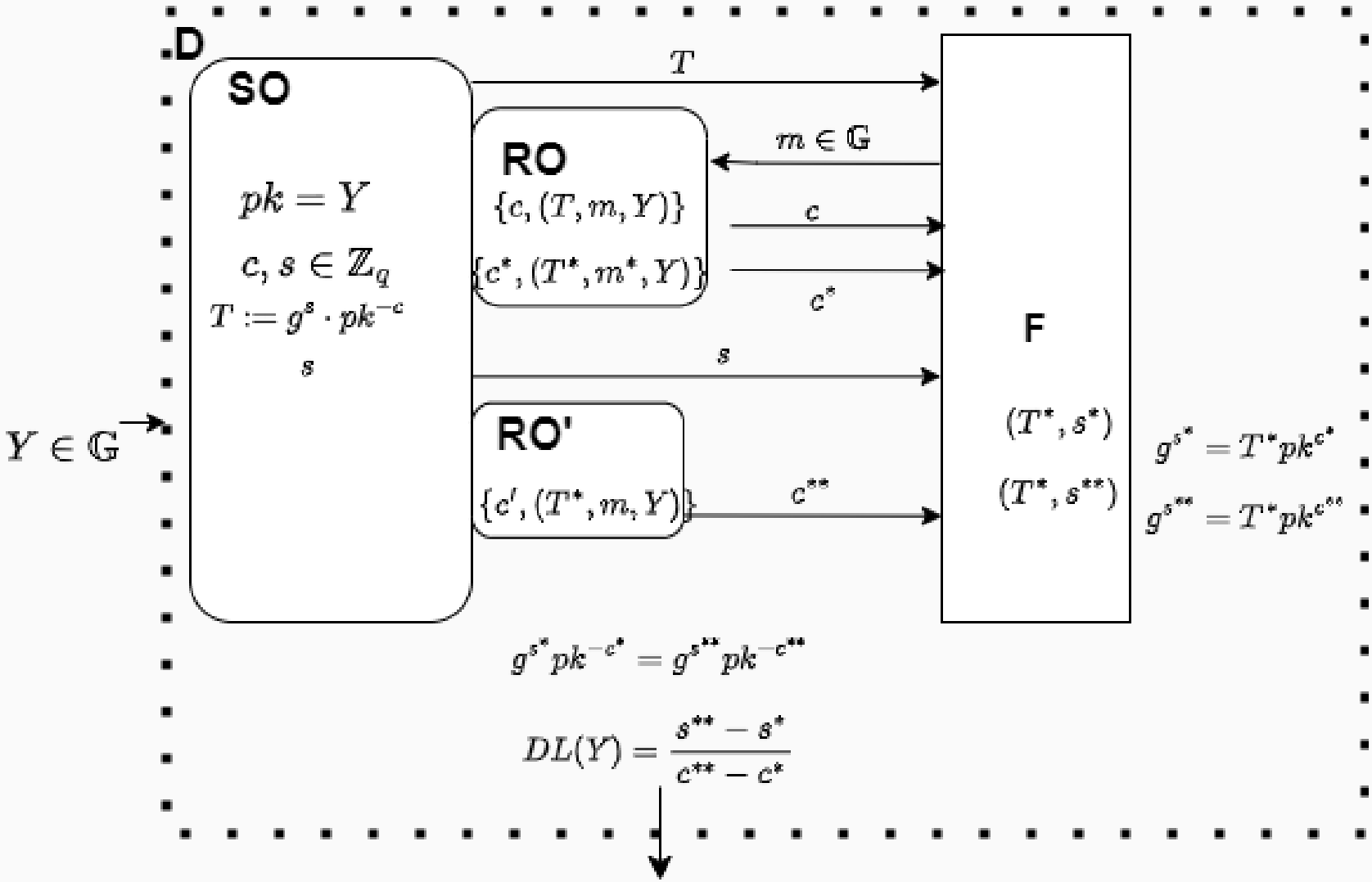
Πατέντα μέχρι το 2008!

Απόδειξη Ασφάλειας Schnorr - [PS00] - Διαισθητικά

- Βασίζεται στην ειδική ορθότητα του Schnorr
- Signing Oracle: χρήση HVZK simulator
- Random Oracle: Για είσοδο (T, m, Y) επιστροφή $c \leftarrow\$ \mathbb{Z}_q$
- Ο $\mathcal{D}$ μαντεύει σε ποιο RO query του $\mathcal{F}$ (T^*, m, Y) θα γίνει το forgery (T^*, s^*) και το απαντάει με $c^* \leftarrow\$ \mathbb{Z}_q$
- Μόλις ο $\mathcal{F}$ βγάλει το forgery, ο $\mathcal{D}$ κάνει REWIND τον F από το σημείο που ρώτησε το (T^*, m, Y)
- Το RO θα είναι διαφορετικό από εδώ και πέρα (**oracle replay attack**)
- Η ερώτηση (T^*, m, Y) θα απαντηθεί με $c^{**} (\neq c^*) \leftarrow\$ \mathbb{Z}_q$
- **Forking Lemma [PS00]**: Με μη αμελητέα πιθανότητα ο $\mathcal{F}$ θα βγάλει ξανά forgery (T^*, s^{**})
- Ο $\mathcal{D}$ θα εξάγει τον διακριτό λογάριθμο

Πλεονέκτημα $\mathcal{D} \in \mathcal{O}\left(\frac{p_F^2}{q_H}\right)$ - λιγότερο tight απόδειξη

Απόδειξη Ασφάλειας Schnorr - [PS00] - Διαισθητικά



Ψηφιακές Υπογραφές - (EC)DSA

Βασικά Στοιχεία

- DSA: NIST, 1991.
- Παράκαμψη Πατέντας Σε Υπογραφή Schnorr
- Παραλλαγή του ElGamal, μικρότερο μέγεθος υπογραφής.
- Αρχικά: λειτουργία σε μια υποομάδα της $\mathbb{Z}_p^*$, τάξης 2^{160}.
- Αργότερα: Υλοποίηση με ελλειπτικές καμπύλες - ECDSA
- Χρήση σε Bitcoin, SSL, SSH ...
- Δεν υπάρχει απόδειξη ασφάλειας!

Δημιουργία κλειδιών

- Δημόσια Διαθέσιμες Παράμετροι: $(p, a, b, \#\mathcal{E}, q, G)$
- Δουλεύουμε σε υποομάδα τάξης q στην καμπύλη $y^2 = x^3 + ax + b, \;\; a, b \in \mathbb{F}_p$ με σημείο βάσης το G
- Ιδιωτικό κλειδί: Ένας τυχαίος ακέραιος $x \in \{1, \cdots, q-1\}$
- Δημόσιο κλειδί: Το σημείο $Y = xG \in \mathcal{E}$

Υπογραφή

- Υπολογισμός σύνοψης του μηνύματος $h := \mathrm{H}(m)$ και προσαρμογή της στο $[0, \cdots, q-1]$
- Επιλογή τυχαίου αριθμού k στο σύνολο $\{1, \cdots, q-1\}$
- Υπολογισμός του σημείου $P := kG = (x_P, y_P)$.
- Υπολογισμός του $r := x_P \bmod q$
- Αν $r = 0 \pmod q$ τότε επανάληψη με καινούριο k.
- Υπολογισμός του $s := (h + r \cdot x)k^{-1} \bmod q$
- Αν $s = 0$ τότε επανάληψη με καινούριο k.
- Η υπογραφή είναι το ζεύγος (r, s)

ECDSA iii

Επαλήθευση

- Υπολογισμός του $h := \mathrm{H}(m)$
- Υπολογισμός του $u_1 := s^{-1}h \bmod q$
- Υπολογισμός του $u_2 := s^{-1}r \bmod q$
- Υπολογισμός του σημείου $P' := u_1G + u_2Y$
- Η υπογραφή είναι έγκυρη αν $r = x_{P'} \pmod q$

Ορθότητα: Υπολογισμός ίδιου σημείου με 2 τρόπους

- Υπογραφή $P = kG$
- Επαλήθευση $P' = u_1G + u_2Y$

$$P' = u_1G + u_2Y = s^{-1}(h + rx)G = k(h + rx)^{-1}(h + rx)G = kG = P$$

ECDSA malleability - όχι strong $\mathrm{EUF\text{-}CMA}$

- Παρατήρηση: Αν (r, s) είναι έγκυρη υπογραφή, τότε και $(r, -s)$ έγκυρη υπογραφή (λόγω συμμετρίας της ελλειπτικής καμπύλης)
- Επίσης:
  $P = u_1 G + u_2 Y \Rightarrow Y = u_1 \cdot u_2^{-1} G - u_2^{-1} P = r^{-1}(\mathrm{H}(m) \cdot G - sP)$
- Δηλαδή: Δημόσιο κλειδί Y συνάρτηση σ, m
- Αν βρω μια έγκυρη υπογραφή, μπορώ να αλλάξω $(r, s), m$ ώστε να βρω μια έγκυρη υπογραφή για διαφορετικό κλειδί

Ασφάλεια: Επιλογή διαφορετικού k ανά υπογραφή

Αλλιώς: Ανάκτηση ιδιωτικού κλειδιού!

Επίθεση επανάληψης τυχαιότητας
Δίνονται δύο υπογραφές $(r_1, s_1)(r_2, s_2)$

Παρατήρηση: $r_1 = r_2 = x_{kG}$

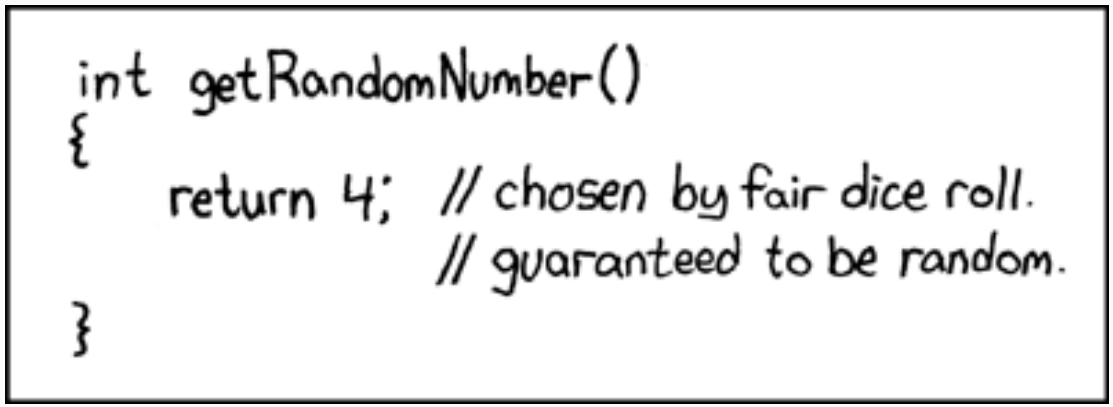
Τότε: $s_1 - s_2 = k^{-1}(h_1 - h_2) \pmod q$

Ανάκτηση $k := (h_1 - h_2)(s_1 - s_2)^{-1} \pmod q$

Ανάκτηση $x := (ks_1 - h_1)r^{-1}$

ECDSA vi

Sony PlayStation 3 hack (2011): Υπογραφή όλων των παιχνιδιών με ίδιο k



https://xkcd.com/221/

ECDSA vii

Περαιτέρω ανάλυση: Αποκάλυψη ή πρόβλεψη ορισμένων bits του $k \Rightarrow$ αποκάλυψη του ιδιωτικού κλειδιού!

Παραδείγματα

- αποκάλυψη 3 bits από κάθε nonce 100 υπογραφών (2003 - 160bit ECDSA)
- αποκάλυψη 5 bits από κάθε nonce 4000 υπογραφών (2013 - 384bit ECDSA)
- 4 bits πρόβλεψη με κάποια εκατομμύρια υπογραφών (2019)
- αποκάλυψη < 1 bit με κάποια εκατομμύρια υπογραφών (2020)

Μία λύση: Deterministic Nonces ή καλύτερα περιορισμός χρήσης του RNG

Ψηφιακές Υπογραφές - edDSA i

Γενικά

- Daniel J. Bernstein, Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang (2012)
- Σε διαδικασία προτυποποίησης από NIST
- Υπογραφές Schnorr σε ελλειπτικές καμπύλες Edwards (λήξη πατέντας Schnorr)
- Ντετερμινιστική παραγωγή τυχαιότητας κατά την υπογραφή
- Χρήση random number generator μόνο κατά την δημιουργία κλειδιών

Ψηφιακές Υπογραφές - edDSA ii

Δημιουργία Κλειδιών

Δημόσια Διαθέσιμες Παράμετροι:

$(q = 2^{255} - 19, \mathcal{E} : -x^2 + y^2 = 1 - \frac{121665}{121666}x^2y^2, G)$

- Επιλογή τυχαιότητας $k \leftarrow\$ \{0,1\}^{256}$
- Χρήση συνάρτησης SHA-512 $h := \mathsf{H}(k)$ με
- $x := h[0..255]$ κλειδί υπογραφής
- $r := h[256..511]$ κλειδί τυχαιότητας
- $\mathsf{pk} := xG, \mathsf{sk} := (x, r)$

Ψηφιακές Υπογραφές - edDSA iii

Δημιουργία Υπογραφής $\mathrm{Sign}(sk, m)$

- $r_m := \mathrm{H}(r||m)$
- $R_m := r_m G$
- $c := \mathrm{H}(\mathrm{pk}, R_m, m)$
- $s := r_m + c \cdot x \bmod q$

Η υπογραφή είναι το (R_m, s)

Ίδιο μήνυμα $\Rightarrow$ ίδια υπογραφή

Επαλήθευση Υπογραφής $\mathrm{Vf}(pk, m, (R_m, s))$
Υπολογισμός: $c' := \mathrm{H}(\mathrm{pk}, R_m, m)$

Έγκυρη ανν: $s \cdot G = R_m + c'\mathrm{pk}$

Αυθεντικότητα Κλειδιών

Πρακτική χρήση ψηφιακών υπογραφών

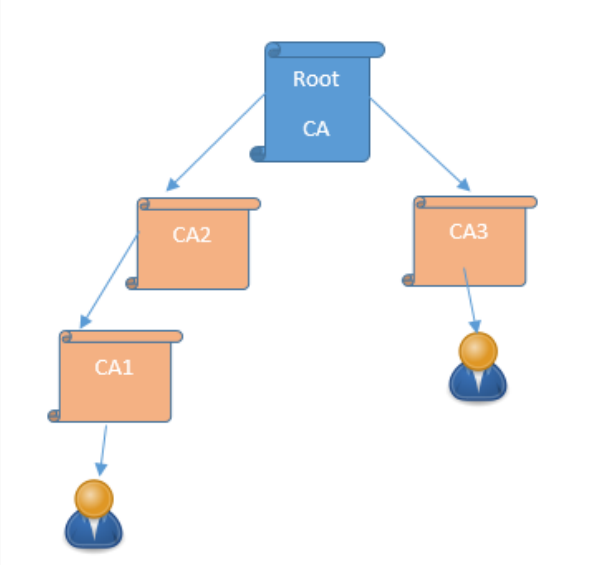
- Διαφορά Συμμετρικών - Ασύμμετρων Κρυπτοσυστημάτων
  - Συμμετρικά: Δύσκολη διανομή, Εύκολη Αυθεντικότητα (λόγω φυσικών υποθέσεων)
  - Ασύμμετρα: Εύκολη διανομή, Δύσκολη Αυθεντικότητα
- Αντιστοιχία (?) Ταυτότητας Χρήστη - Δημοσίου, Ιδιωτικού Κλειδιού (binding)
- Ενεργός αντίπαλος - Πλαστοπροσωπία - αλλαγή κλειδιών
- Απαραίτητη η διασφάλιση για χρήση σε ευρεία κλίμακα
- Δεν υπάρχει λύση που να δουλεύει θεωρητικά **και** πρακτικά
- Στην πράξη: μετάθεση του προβλήματος με μείωση της έκτασης (αρκεί 1 αυθεντικό κλειδί)

Αρχές Πιστοποίησης (Certification Authorities - CAs)

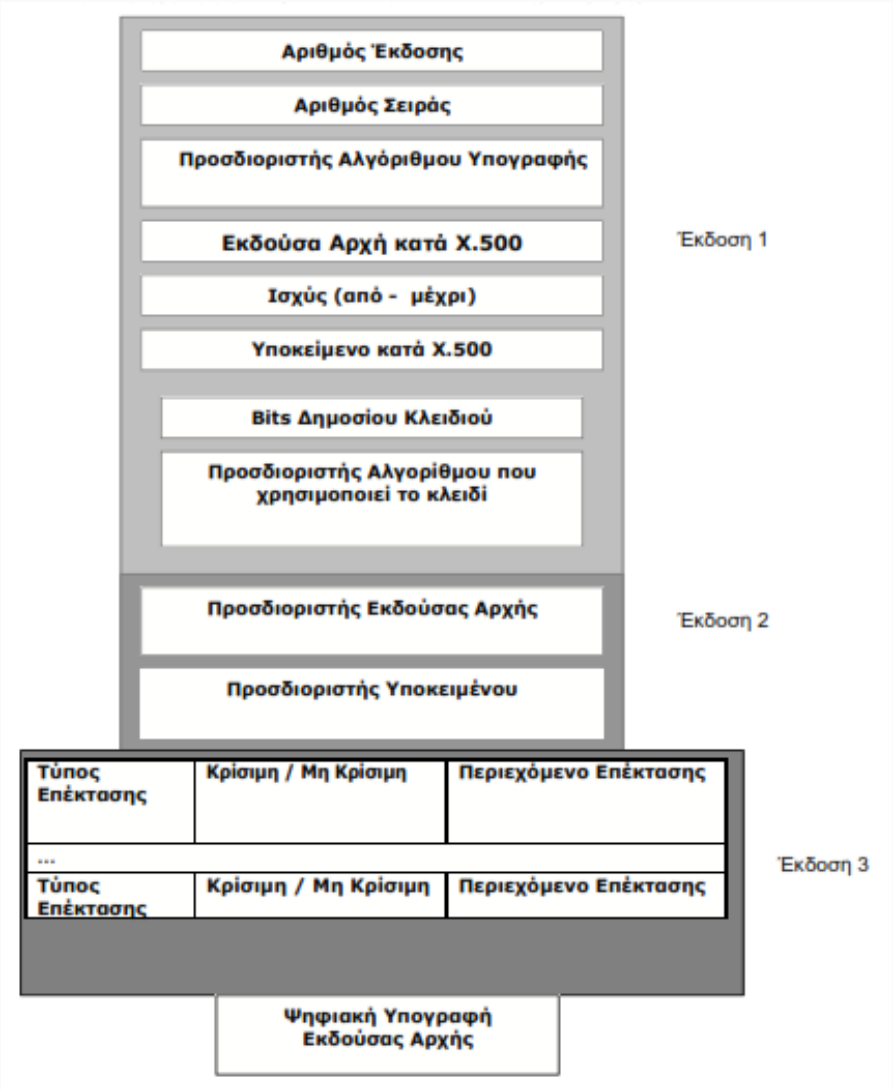
- *Έμπιστες* Τρίτες Οντότητες - (Πάροχοι Υπηρεσιών Πιστοποίησης)
  - Πιστοποίηση Αντιστοιχίας Ταυτότητας Κλειδιών
  - Εγγυάται ότι το δημόσιο κλειδί *όντως* αντιστοιχεί στον χρήστη
  - Πώς; Υπογράφοντας 'ψηφιακά' το ζεύγος (ID, PK_{ID})
- Πλεονέκτημα: Μείωση κλειδιών που πρέπει να αποκτήσουμε με έμπτιστο τρόπο
  - Μόνο το κλειδί της CA
  - Για τα υπόλοιπα 'εγγύαται' το πιστοποιητικό
- Μειονέκτημα Ποιος εγγυάται την σχέση κλειδιών-ταυτότητας για την CA;
  - Η ίδια! (υπογράφει η ίδια μία δήλωση για τον εαυτό της)
  - ή μια άλλη *ανώτερη* αρχή πιστοποίησης!

Υποδομή Δημοσίου Κλειδιού

- Ιεραρχική οργάνωση δομών πιστοποίησης
- Οργάνωση των αρχών πιστοποίησης και των σχετικών υπηρεσιών
- Loren Kohnfelder, MIT BSc thesis, 1978
- Ευρεία προτυποποίηση (ITU X.500, RFC 6818)
- Ενδιάμεσες Αρχές: Υπογραφή από ανώτερη αρχή
- Ριζικές (Root) Αρχές: Υπογράφουν μόνες τους
- Συνήθως 3-4 επίπεδα



Πιστοποιητικό X.509



Απόκτηση πιστοποιητικών

- Προεγκατάσταση στο λειτουργικό σύστημα
- Προεγκατάσταση στον περιηγητή
- Απόκτηση από αρχείο/ιστοσελίδα
- Απόκτηση από νομική οντότητα (εταιρεία, κράτος)

Supply Chain Attacks

Αρχές Πιστοποίησης - Άλλες υπηρεσίες

- Διάδοση Πιστοποιητικών σε αποθετήρια
- Εγγραφή-Επαλήθευση Ταυτότητας Χρηστών
- Δημιουργία κρυπτογραφικών κλειδιών (αυστηρές προδιαγραφές ασφάλειας)
- Ανάκληση Πιστοποιητικών - Ενημέρωση
- Χρονοσήμανση - Αρχειοθέτηση

Ανάκληση Πιστοποιητικών

Άκυρα πιστοποιητικά

- Απώλεια κλειδιού υπογραφής, Αλλαγή Στοιχείων Υποκειμένου,
- Ενημέρωση Χρηστών με 2 τρόπους
- Certificate Revocation Lists (CRL):
  - ‘Μαύρη‘ λίστα από SN για πιστοποιητικά που δεν ισχύουν
  - Υπογεγραμμένη από την CA
  - Ανάκτηση σε τακτά χρονικά διαστήματα
  - Πεδίο CDP
- OCSP (Online Certificate Status Protocol)
  - Ερώτηση στην CA για ισχύ πιστοποιητικού
  - Η CA συμμετέχει σε κάθε συναλλαγή

Εναλλακτικές Προσεγγίσεις: web of trust

Ομότιμη έκδοση και επαλήθευση ταυτότητας (web of trust)

- Κάθε χρήστης είναι CA
- Υπογράφει αντιστοιχίες που γνωρίζει
- Λήψη πιστοποιητικών μόνο από γνωστούς χρήστες
- Ο κάθε χρήστης ‘εγγυάται‘ για τους γνωστούς του
- PGP

Identity based cryptography

- Signatures: Shamir 1984
- Encryption: Boneh-Franklin (2001)
- Οποιοδήποτε όνομα κάποιου χρήστη πχ. email είναι η ταυτότητα
- Δεν χρειάζεται διανομή κλειδιού
- Χρειάζεται κεντρική TTP
- Παράγει τα ιδιωτικά κλειδιά από την ταυτότητα

Identity based signatures

- TTP έχει κλειδί RSA $((e, n), d)$
- Δημιουργία ιδιωτικού κλειδιού από ταυτότητα χρήστη id
  - Υπογραφή σύνοψης της ταυτότητας
  - $k = \mathrm{H}(id)^d \bmod n$
  - Ασφαλής Διανομή στον κάτοχο
- Υπογραφή από χρήστη id
  - Επιλογή τυχαίου r
  - $t = r^e \bmod n$
  - $s = k\, r^{\mathrm{H}(m,t)} \bmod n$
  - Η υπογραφή είναι (t, s)
- Επαλήθευση υπογραφής με την ταυτότητα:
- Έλεγχος αν: $\mathrm{H}(id)t^{\mathrm{H}(m,t)} = s^e$
- Ορθότητα: $s^e = k^e r^{e\mathrm{H}(m,t)} = \mathrm{H}(id)t^{\mathrm{H}(m,t)}$

Υπογραφές με ιδιωτικότητα

Motivation

Ψηφιακές Υπογραφές: Ακεραιότητα, Αυθεντικότητα, Μη - Αποκήρυξη, Δημόσια επαληθευσιμότητα

Χωρίς ιδιωτικότητα όμως...

- Ο $\mathcal{S}$ βλέπει το μήνυμά που υπογράφει και
- μπορεί να συσχετίσει την υπογραφή με το αίτημα δημιουργίας της
- Επίσης: Το δημόσιο κλειδί επαλήθευσης προδίδει τον signer.
- Τα παραπάνω δεν είναι επιθυμητά σε πολλές εφαρμογές
  - Ηλεκτρονικό χρήμα
  - Ηλεκτρονικές ψηφοφορίες
  - Whistleblowing

Υπογραφές με ιδιωτικότητα

Τυφλές υπογραφές για Ηλεκτρονικό χρήμα με TTP

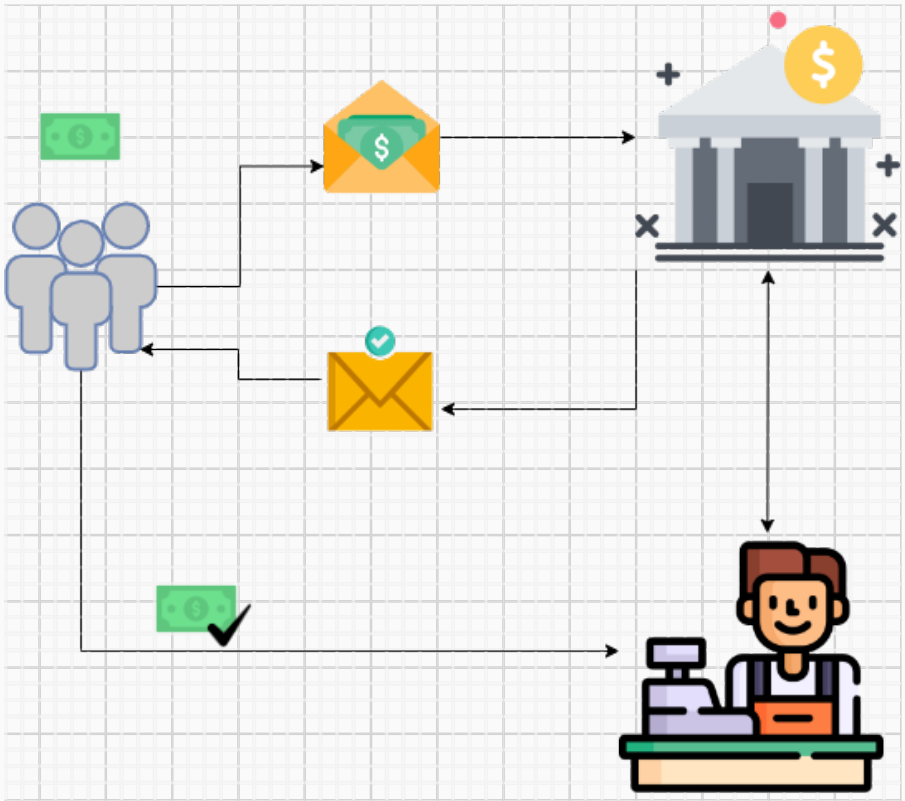
- Νόμισμα $c \leftarrow\!\!\$\ \{0,1\}^*$ με συγκεκριμένη αξία (πχ. 1).
- Για ασφάλεια (αποφυγή double, overspending): υπογραφή από τράπεζα.
- Ο **Αγοραστής** ζητάει από την **Τράπεζα** ένα νόμισμα c.
- Ο **Αγοραστής** αγοράζει κάτι από τον **Πωλητή** με το c.
- Ο **Πωλητής** επικοινωνεί με την τράπεζα για να βεβαιώσει ότι το c δεν έχει ξαναξοδευτεί.
  Αν δεν έχει ξαναξοδευτεί το δέχεται και ολοκληρώνει τη συναλλαγή.
- Η **Τράπεζα** μαρκάρει το νόμισμα c ως ξοδεμένο.
- Αργότερα ο **Πωλητής** παίρνει από την τράπεζα την αξία του c.

Όμως: Η **Τράπεζα** γνωρίζει πού ξοδεύτηκε το νόμισμα

Ανώνυμο Ηλεκτρονικό χρήμα (με TTP)

Λύση: Φάκελος με καρμπόν

- Το νόμισμα μπαίνει σε φάκελο
- Η **Τράπεζα** υπογράφει το φάκελο
- Το καρμπόν μεταφέρει την υπογραφή στο νόμισμα
- Το νόμισμα βγαίνει από τον φάκελο πριν ξοδευτεί
- Η **Τράπεζα** δεν μπορεί να συσχετίσει νόμισμα με φάκελο



Η υπογραφή είναι τυφλή.

Σύνταξη τυφλών υπογραφών

Ένα σχήμα τυφλών υπογραφών είναι μια τριάδα $\Pi = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$:

- $(\mathsf{sk}, \mathsf{vk}, \mathsf{prms}) \leftarrow \mathsf{KGen}(1^\lambda)$
  Δημιουργία κλειδιών και κρυπτογραφικών παραμέτρων
- $\sigma \leftarrow \mathsf{Sign}\langle \mathcal{S}(\mathsf{sk}), \mathcal{U}(m), \mathsf{vk} \rangle$
  Το Sign είναι *πρωτόκολλο* και όχι αλγόριθμος. Συνήθως:
  - $m' := \mathsf{Blind}(m, \mathsf{vk})$ εκτελείται από τον $\mathcal{U}$
  - $\sigma' := \mathsf{Sign}(m', \mathsf{sk})$ όπου ο $\mathcal{S}$ εκτελεί τον *αλγόριθμο* Sign
  - $\sigma := \mathsf{Unblind}(\sigma', \mathsf{vk})$ εκτελείται από τον $\mathcal{U}$
- Επαλήθευση:
  $\{0, 1\} \leftarrow \mathsf{Vf}(m, \sigma, \mathsf{vk})$

Ορθότητα:
$\mathsf{Vf}(m, \mathsf{Sign}\langle \mathcal{S}(\mathsf{sk}), \mathcal{U}(m), \mathsf{vk} \rangle, \mathsf{vk}) = 1$ για
$(\mathsf{sk}, \mathsf{vk}, \mathsf{prms}) \leftarrow \mathsf{KGen}(1^\lambda)$

Ιδιωτικότητα

Ο αντίπαλος είναι ο υπογράφων $\mathcal{S}$.

- Δεν πρέπει να μάθει τίποτα για το μήνυμα που υπογράφει
- Βλέποντας μήνυμα και υπογραφή να μην μπορεί να το συσχετίσει με κάποια εκτέλεση του Sign.

Algorithm 1: $\mathsf{BlindGame}_{\Pi,\mathcal{A}}$

Input : λ

Output: $\{0,1\}$

$(\mathsf{sk},\mathsf{vk},\mathsf{prms}) \leftarrow \mathsf{KGen}(1^\lambda)$

$(m_0,m_1) \leftarrow \mathcal{A}(\textbf{find},1^\lambda)$

$b \leftarrow\!\!\$\ \{0,1\}$

$\sigma_b \leftarrow \mathsf{Sign}\langle \mathcal{A}(\textbf{issue},\mathsf{sk}),\mathcal{U}(m_b),\mathsf{vk}\rangle$

$\sigma_{1-b} \leftarrow \mathsf{Sign}\langle \mathcal{A}(\textbf{issue},\mathsf{sk}),\mathcal{U}(m_{1-b}),\mathsf{vk}\rangle$

if $\mathsf{Vf}(m_b,\sigma_b,\mathsf{vk}) = 1$ AND $\mathsf{Vf}(m_{1-b},\sigma_{1-b},\mathsf{vk}) = 1$ **then**

 | $b' \leftarrow \mathcal{A}(\textbf{guess},\sigma_0,\sigma_1)$

end

return $b = b'$

Ο αντίπαλος πρέπει να μαντέψει τη σειρά υπογραφής.

Blindness

Perfect Blindness

Ένα σχήμα τυφλών υπογραφών Π είναι **τέλεια μυστικό** αν *για κάθε αντίπαλο* $\mathcal{A}$ ισχύει ότι

$$\Pr[\text{BlindGame}_{\Pi,\mathcal{A}}(1^\lambda) = 1] = \frac{1}{2}$$

Computational Blindness

Ένα σχήμα τυφλών υπογραφών Π είναι **υπολογιστικά μυστικό** αν *για κάθε PPT αντίπαλο* $\mathcal{A}$ ισχύει ότι

$$\Pr[\text{BlindGame}_{\Pi,\mathcal{A}}(1^\lambda) = 1] \leq \frac{1}{2} + \text{negl}(\lambda)$$

Unforgeability για τυφλές υπογραφές

- Δεν έχει νόημα το μοντέλο των κλασικών υπογραφών (EUF-CMA).
- Εξήγηση:
  - Ο $\mathcal{S}$ δημιούργησε (m', σ')
  - Ο $\mathcal{U}$ από αυτό έφτιαξε (m, σ)
  - ... για το οποίο $\mathsf{Vf}(m, \sigma, \mathsf{vk}) = 1$
  - Δηλ. ο $\mathcal{U}$ έφτιαξε έγκυρη υπογραφή χωρίς να έχει ιδιωτικό κλειδί (έκανε πλαστογράφηση)

Ορίζουμε το unforgeability με βάση το σενάριο χρήσης του e-cash.

Ο αντίπαλος (τώρα ο χρήστης) δεν μπορεί να φτιάξει περισσότερα νομίσματα από αυτά που του έδωσε η τράπεζα.

Νέα έννοια unforgeability: **One-more unforgeability**

One-more unforgeability

Algorithm 2: $\mathsf{OneMoreForge}_{\mathcal{A},\Pi}$

Input : λ

Output: $\{0,1\}$

$(\mathsf{sk},\mathsf{vk},\mathsf{prms}) \leftarrow \mathsf{KGen}(1^\lambda)$

$\{(m_i,\sigma_i)\}_{i=1}^{l+1} \leftarrow \mathsf{Sign}\langle \mathcal{S}(\mathsf{sk}), \mathcal{A}(m_j), \mathsf{vk}\rangle_{j=1}^{k}$

if $(\forall i,j \in [l+1]$ με $i \neq j \Rightarrow m_i \neq m_j)$ AND

$(\forall i \in [l+1]\, \mathsf{Vf}(m_i,\sigma_i,\mathsf{vk}) = 1)$ AND

$k \leq l$ **then**

  return 1

else

  return 0

end

l: Το μέγιστο πλήθος των sessions $\langle \mathcal{S}, \mathcal{A}\rangle$. Μπορούν να είναι σειριακά ή παράλληλα!

One-more unforgeability

Definition
Ένα σχήμα τυφλών υπογραφών είναι **One-More Unforgeable** αν για κάθε PPT αντίπαλο $\mathcal{A}$ που εκτελεί το πολύ poly(λ) πρωτόκολλα Sign ισχύει ότι

$$\Pr[\text{OneMoreForge}_{\Pi,\mathcal{A}}(1^\lambda) = 1] \leq \text{negl}(\lambda)$$

Definition
Ένα σχήμα τυφλών υπογραφών είναι **Strongly One-More Unforgeable** αν για κάθε PPT αντίπαλο $\mathcal{A}$ που εκτελεί το πολύ poly$log(\lambda)$ πρωτόκολλα Sign ισχύει ότι

$$\Pr[\text{OneMoreForge}_{\Pi,\mathcal{A}}(1^\lambda) = 1] \leq \text{negl}(\lambda)$$

RSA Blind Signatures

- $((n, e), d) \leftarrow \mathsf{KGen}(1^\lambda)$
- $(m', r) := \mathsf{Blind}(m, (n, e))$
  $r \leftarrow\!\!\$\ \mathbb{Z}_n^*$
  $m' \leftarrow \mathsf{H}(m) \cdot r^e \bmod n$
- $\sigma' \leftarrow \mathsf{Sign}(m', d, (n, e))$
  $\sigma' \leftarrow m'^d \bmod N$
- $\sigma \leftarrow \mathsf{Unblind}(\sigma', r, (n, e))$
  $\sigma \leftarrow \sigma' \cdot r^{-1} \bmod N$
- $\mathsf{Vf}(m, \sigma, (n, e)) = 1 \Leftrightarrow \sigma^e \equiv \mathsf{H}(m) \pmod n$

Ορθότητα

$$\begin{aligned}\sigma^e &\equiv (\sigma' \cdot r^{-1})^e \equiv (m'^d \cdot r^{-1})^e \\ &\equiv ((\mathsf{H}(m) \cdot r^e)^d \cdot r^{-1})^e \\ &\equiv (\mathsf{H}(m)^d \cdot r \cdot r^{-1})^e \\ &\equiv \mathsf{H}(m) \pmod n\end{aligned}$$

και ο $\mathcal{V}$ αποδέχεται.

RSA - Blindness

Τυφλότητα (Διαισθητικά)

Κάθε υπογραφή εξαρτάται από m, r - Μία σχέση - δύο άγνωστοι!

Ισχύει ότι $m' \equiv \mathrm{H}(m) r^e \mod n$ δηλαδή $r^e \equiv m' \mathrm{H}(m)^{-1} \pmod n$

Έγκυρη σ για κάθε m με κατάλληλη επιλογή r!

Θεώρημα

Οι υπογραφές RSA παρέχουν perfect blindness

Στο BlindGame ο αντίπαλος βλέπει:

$\mathrm{view}_i = (m'_i, \sigma'_i)$, σ_j για $i, j \in \{0, 1\}$

Σε κάθε περίπτωση υπάρχει μοναδικό r ώστε view_i να αντιστοιχεί στο σ_j

Συγκεκριμένα $r = \sigma'_i \cdot \sigma_j^{-1}$

Άρα η καλύτερη στρατηγική του $\mathcal{A}$ είναι να μαντέψει στην τύχη.

RSA - Unforgeability

Algorithm 3: RSA-CTI πρόβλημα [BNPS 2003]

Input : λ

Output: $\{0,1\}$

$(d,(e,n)) \leftarrow \mathsf{KGen}(1^\lambda)$

for $i := 1$ **to** m **do**

  $y_i \leftarrow\!\!\$\ \mathbb{Z}_n^*$

end

$(\pi, \{x_i\}_{i=1}^{l+1}) \leftarrow \mathcal{A}^{(\cdot)^d}(n,e,\{y_i\}_{i=1}^m)$ (k ερωτήσεις)

if $\pi : [l+1] \to [n]$ *είναι 1-1* AND

  $\forall i \in [l+1] : x_i^e = y_{\pi(i)}$ AND $k \leq l$ **then**

    return 1

else

    return 0

end

Θεώρημα

Οι υπογραφές RSA παρέχουν one-more unforgeability στο μοντέλο του τυχαίου μαντείου αν το πρόβλημα RSA-CTI είναι δύσκολο