ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΚΡΥΠΤΟΛΟΓΙΑ
ΣΗΜΕΙΩΣΕΙΣ #9

ΘΕΟΔΟΥΛΟΣ ΓΑΡΕΦΑΛΑΚΗΣ

1. Ψηφιακες Υπογραφες

Μια από τις σημαντικότερες εφαρμογές της κρυπτογραφίας δημόσιου κλειδιού είναι ο ψηφιακές υπογραφές. Είναι το μαθηματικό/αλγοριθμικό ανάλογο των κοινών υπογραφών. Πολύ γενικά, σε ένα σύστημα ψηφιακών υπογραφών κάθε χρήστης έχει ένα ζεύγος δημόσιου/ιδιωτικού κλειδιών (K_p, K_s). Το σύστημα αποτελείται από δύο αλγόριθμους:

- Τον αλγόριθμο υπογραφής S, ο οποίος δεδομένου ενός μηνύματος m και ενός ιδιωτικού κλειδιού K_s υπολογίζει την υπογραφή $s = S_{K_s}(m)$.
- Τον αλγόριθμο πιστοποίησης V, ο οποίος δεδομένης μίας υπογραφής s, ενός μυνήματος m και ενός δημόσιου κλειδιού K_p απαντά «ΝΑΙ» αν $s = S_{K_s}(m)$ και «ΟΧΙ» διαφορετικά.

Όπως και στις κοινές υπογραφές, ένα σύστημα ψηφιακών υπογραφών πρέπει να έχει τις παρακάτω βασικές ιδιότητες:

(1) Οι αλγόριθμοι S και V είναι πολυωνυμικού χρόνου, δηλαδή κάθε χρήστης μπορεί εύκολα να υπογράψει ένα μήνυμα της επιλογής του και να πιστοποιήσει μια υπογραφή που έβαλε οποιοσδήποτε άλλος χρήστης σε οποιοδήποτε μήνυμα.

(2) Είναι υπολογιστικά ανέφικτο κάποιος να παράγει μια υπογραφή s σε ένα μήνυμα m *χωρίς* να γνωρίζει το ιδιωτικό κλειδί K_s για την οποία ο αλγόριθμος πιστοποίησης V με δεδομένα τα K_p (το δημόσιο κλειδί που αντιστοιχεί στο ιδιωτικό κλειδί K_s) και το m να απαντά «ΝΑΙ». Δηλαδή είναι ανέφικτο κάποιος να πλαστογραφεί την υπογραφή κάποιου άλλου σε οποιοδήποτε μήνυμα.

2. Ασφαλεια

Προκειμένου να ορίσουμε την έννοια της ασφάλειας για ψήφιακές υπογραφές, πρέπει να εξετάσουμε δύο πράγματα: πρώτον τις δυνατότητες που έχει ο επιτηθέμενος (αυτός δηλαδή που επιχειρεί να πραγματοποιήσει μια πλαστογραφία) και δεύτερον πότε θεωρούμε ότι ο επίδοξος πλαστογράφος έχει επιτύχει.

Επιθέσεις επιλεγόμενων μηνυμάτων. Ας δούμε τι δυνατότητες πα ραχωρούμε στον επιτηθέμενο. Κατ' αρχάς, όπως σε κάθε σύστημα δημόσιου κλειδιού πρέπει να περιορίσουμε τον επιτηθέμενο σε εφικτούς υπολογισμούς. Τυπικά αυτό σημαίνει ότι είναι ένας αλγόριθμος πολυωνυμικού χρόνου. Κατά τ' άλλα τον διευκολύνουμε όσο το δυνατό περισσότερο: του επιτρέπουμε να ζητήσει να δει υπογραφές $s_1, \ldots, s_t$ σε μια σειρά από μηνύματα της επιλογής του $m_1, \ldots, m_t$.

Οι υπογραφές στα μηνύματα έχουν υπολογιστεί με χρήση του ιδιωτικού κλειδιού K_s στο οποίο γίνεται η επίθεση και βέβαια οι υπογραφές είναι ισχύουσες. Παρατηρήστε ότι με το να παρέχουμε στον επιτηθέμενο ζεύγη (s_i, m_i) μηνυμάτων, υπογραφών μόνο βοήθεια του προσφέρουμε, καθώς τα ζεύγη αυτά ίσως του δώσουν μια ιδέα για το πώς ένα μήνυμα συνδέεται με μια υπογραφή. Όπως θα δούμε παρακάτω αυτή η βοήθεια είναι αρκετή για απρόσεχτα σχεδιασμένα συστήματα.

Υπαρκτή πλαστογραφία. Περνάμε τώρα στην έννοια της πλαστογραφίας. Στόχος του επιτηθέμενου (στο κλειδί K_s) είναι προφανώς να παράγει ένα ζεύγος (s, m) όπου το s είναι μια ισχύουσα υπογραφή στο μήνυμα m, δηλαδή $V_{K_p}(m, s) =$«ΝΑΙ», όπου K_p είναι το δημόσιο κλειδί που αντιστοιχεί στο ιδιωτικό κλειδί K_s. Είναι ακόμα φανερό ότι το m δε μπορεί να είναι ένα εκ των $m_1, \ldots, m_t$ στα οποία ο επιτηθέμενος έχει *δει* υπογραφές. Του επιτρέπουμε να επιχείρησει να παράγει μία υπογραφή σε οποιοδήποτε άλλο μήνυμα επιθυμεί. Το μήνυμα m δηλαδή μπορεί να είναι οτιδήποτε (εκτός των $m_1, \ldots, m_t$) ακόμα και αν δεν έχει κάποιο νόημα σε κάποια ανθρώπινη γλώσσα.

Ασφάλεια. Ονομάζουμε ένα σύστημα ασφαλές (ή πιο σωστά, ασφαλές ενάντια σε υπαρκτές πλαστογραφίες με επιθέσεις επιλεγόμενων μηνυμάτων), αν δεν υπάρχει πολυωνυμικού χρόνου αλγόριθμος, ο οποίος, δεδομένου ενός δημόσιου κλειδιού K_p, παράγει μια υπαρκτή πλαστογραφία μετά από επίθεση επιλεγόμενων μηνυμάτων με πιθανότητα μη αμελητέα.

3. Υπογραφες RSA

Στη δημοσίευση του 1978, μαζί με το σύστημα κρυπτογράφησης, οι Rivest, Shamir και Adleman πρότειναν ένα σύστημα ψηφιακών υπογραφών, τις οποίες θα ονομάζουμε υπογραφές RSA. Οι παράμετροι του συστήματος υπογραφών είναι ακριβώς οι ίδιες με αυτές του συστήματος κρυπτογράφησης. Τις επαναλαμβάνουμε σύντομα. Κάθε χρήστης επιλέγει δύο πρώτους p, q και υπολογίζει το $n = pq$. Υπολογίζει το $\phi(n) = (p-1)(q-1)$ και επιλέγει ένα ακέραιο e πρώτο προς τον $\phi(n)$. Έπειτα υπολογίζει τον αντίστροφο του e modulo n, δηλαδή το d που ικανοποιεί την $ed \equiv 1 \pmod{\phi(n)}$. Δημοσιοποιεί το (n, e) που είναι το δημόσιο κλειδί και κράτα το d κρυφό (είναι το ιδιωτικό κλειδί). Επιτρεπτά μηνύματα είναι ακέραιοι $1 \leq m < n$ με $(m, n) = 1$.

Αλγόριθμος υπογραφής

(1) Δεδομένων του μηνύματος m και του ιδιωτικού κλειδιού d, υπολογίζει το $s = m^d \mod n$.
(2) Η υπογραφή στο μήνυμα m είναι το s.

Αλγόριθμος πιστοποίησης

(1) Δεδομένων του μηνύματος m, της υπογραφής s και του δημόσιου κλειδιού (e, n), υπολογίζει το $m' = s^e \mod n$.
(2) Αν $m' = m$ απαντά «ΝΑΙ», διαφορετικά απαντά «ΟΧΙ».

3.1. **Ασφάλεια του βασικού RSA.** Όπως και στο σύστημα κρυπτογράφησης RSA, οποιοσδήποτε μπορεί να παραγοντοποιεί ακεραίους γρήγορα (δηλαδή σε πολυωνυμικό χρόνο) μπορεί να παραγοντοποιήσει το n και μετά να υπολογίσει το d όπως ακριβώς και ο νόμιμος κάτοχος. Επίσεις, οποιοσδήποτε μπορεί να λύσει το πρόβλημα RSA, δηλαδή δεδομένων των n, e και κάποιου $1 \leq m < n$ με $(m, n) = 1$ μπορεί να υπολογίσει το $m^d \mod n$, μπορεί να πλαστογραφεί υπογραφές σε οποιοδήποτε μήνυμα.

Μπορεί κανείς να πλαστογραφεί αν το πρόβλημα RSA (και συνεπώς η παραγοντοποίηση ακεραίων) είναι υπολογιστικά δύσκολο; Ας δούμε σε μερικά παραδείγματα υπαρξιακών πλαστογραφιών με επιθέσεις επιλεγόμενων μηνυμάτων.

Παράδειγμα 3.1. Στο παράδειγμα αυτό θα δούμε πως μπορεί κανείς να επιτύχει πλαστογραφία σε μήνυμα *της επιλογής του* αφού δει υπογραφές σε δύο μηνύματα της επιλογής του. Έστω $m \in (\mathbb{Z}/n\mathbb{Z})^\times$ το μήνυμα στο οποίο θέλουμε να πραγματοποιήσουμε πλαστογραφία. Επιλέγουμε τυχαία ένα $m_1 \in (\mathbb{Z}/n\mathbb{Z})^\times$ και υπολογίζουμε το m_2 το οποίο ικανοποιεί την $m \equiv m_1 m_2 \pmod n$ (υποδείξτε πώς μπορεί κανείς να υπολογίσει το m_2). Στη συνέχεια ζήτουμε και βλέπουμε τις υπογραφές s_1 και s_2 στα μηνύματα m_1 και m_2 αντίστοιχα. Εφόσον οι υπογραφές είναι ισχύουσες, έχουμε

$$\begin{aligned} s_1^e &\equiv m_1 \pmod n \\ s_2^e &\equiv m_2 \pmod n. \end{aligned}$$

Έτσι βλέπουμε ότι το $s = s_1 s_2 \mod n$ ικανοποιεί

$$s^e \equiv s_1^e s_2^e \equiv m_1 m_2 \equiv m \pmod n$$

και συνεπώς είναι μια ισχύουσα υπογραφή στο m.

4. Συναρτησεις κατακερματισμου

Οι συναρτήσεις κατακερματισμού (hash functions) έχουν πολλές εφαρμογές στην επιστήμη υπολογιστών. Για εφαρμογές στην κρυπτογραφία χρειαζόμαστε συναρτήσεις με εξειδικευμένες ιδιότητες προκειμένου να εξασφαλιστεί η ασφάλεια των συστημάτων. Γενικά μια συνάρτηση κατακερματισμού είναι μια συνάρτηση

$$h : X \longrightarrow Y,$$

όπου το πεδίο ορισμού X μπορούμε να θεωρούμε ότι είναι το $\mathbb{Z}$. Το πεδίο τιμών Y είναι πεπερασμένο και εξαρτάται από το συγκεκριμένο κρυπτογραφικό σύστημα. Για το RSA μπορούμε να θεωρούμε ότι είναι το $\{m \in \mathbb{Z} \mid 1 \leq m < n,\ (m, n) = 1\}$, δηλαδή

$$h : \mathbb{Z} \longrightarrow \{m \in \mathbb{Z} \mid 1 \leq m < n,\ (m, n) = 1\}.$$

Μια κρυπτογραφική συνάρτηση κατακερματισμού $h : X \longrightarrow Y$, για να είναι χρήσιμη πρέπει κατ' αρχάς να μπορεί να υπολογιστεί εύκολα. Επιπλέον πρέπει να έχει τις παρακάτω τρεις ιδιότητες.

(1) Ανθεκτική στην εύρεση προεικόνων: Δεδομένου ενός $y \in Y$, είναι υπολογιστικά ανέφικτος ο υπολογισμός ενός $x \in h^{-1}(y)$.

(2) Ανθεκτική στην εύρεση δεύτερης προεικόνας: Δεδομένου ενός $x \in X$ είναι ανέφικτος ο υπολογισμός ενός $x' \in X$ τέτοιου ώστε $h(x) = h(x')$.
(3) Ανθεκτική στην εύρεση συγκρούσεων: Είναι ανέφικτος ο υπολογισμός ενός ζεύγους $(x, x') \in X^2$ τέτοιου ώστε $h(x) = h(x')$.

Ας δούμε πώς χρησιμοποιούμε μια κρυπτογραφική συνάρτηση κατακερματισμού σε συνδοιασμό με το βασικό συστημα RSA που περιγράψαμε παραπάνω.

Αλγόριθμος υπογραφής

(1) Δεδομένων του μηνύματος m και του ιδιωτικού κλειδιού d, υπολογίζει το $s = h(m)^d \mod n$.
(2) Η υπογραφή στο μήνυμα m είναι το s.

Αλγόριθμος πιστοποίησης

(1) Δεδομένων του μηνύματος m, της υπογραφής s και του δημόσιου κλειδιού (e, n), υπολογίζει το $h' = s^e \mod n$.
(2) Υπολογίζει το $h(m)$.
(3) Αν $h' = h(m)$ απαντά «ΝΑΙ», διαφορετικά απαντά «ΟΧΙ».

Τώρα μπορούμε να δούμε τι σκοπούς εξυπηρετούν οι ιδιότητες που απαιτήσαμε να έχει η συνάρτηση h. Αν η ιδιότητα (1) δεν ικανοποιήται, τότε κανείς μπορεί να πραγματοποιήσει πλαστογραφία ως εξης: Επιλέγει m_1, m_2 και ζητά να δει υπογραφές στα m_1, m_2. Έτσι μαθαίνει s_1, s_2 με

$$\begin{aligned} s_1 &\equiv h(m_1) \pmod n \\ s_2 &\equiv h(m_2) \pmod n. \end{aligned}$$

Άρα το $s = s_1 s_2 \mod n$ είναι υπογραφή σε οποιοδήποτε μήνυμα $m \in h^{-1}(h(m_1)h(m_2))$ το οποίο μπορεί να υπολογίσει αφού υποθέσαμε ότι μπορεί να υπολογίζει προεικόνες της h.

Αν η ιδιότητα (2) δεν ικανοποιήται, τότε κανείς μπορεί να πραγματοποιήσει πλαστογραφία *σε μήνυμα της επιλογής του*. Έστω ότι m είναι το μήνυμα στο οποίο θέλει να πλαστογραφήσει μια υπογραφή. Υπολογίζει μία δεύτερη προεικόνα του $h(m)$, δηλαδή υπολογίζει ένα m' τέτοιο ώστε $h(m) = h(m')$ και ζητά να δει την υπογραφή s στο m'. Τότε το s είναι ισχύουσα υπογραφή και στο m.

Αν η ιδιότητα (3) δεν ικανοποιήται, τότε κανείς μπορεί να παράγει μια υπαρκτή πλαστογραφία (όχι πλέον σε μήνυμα της επιλογής του) με τον ίδιο ακριβώς τρόπο όπως στην προηγούμενη παράγραφο.

Υπάρχουν συναρτήσεις κατακερματισμού με τις παραπάνω ιδιότητες; Αν και δε μπορούμε να το αποδείξουμε, πιστεύουμε ότι η απάντηση είναι ναι. Υπάρχουν διάφορες τέτοιες συναρτήσεις, για παράδειγμα οι SHA-1, MD-4, MD-5, οι οποίες λειτουργούν λίγο-πολύ όπως οι τμηματικοί κώδικες και για τις οποίες κανείς δε γνωρίζει πώς να υπολογίζει προεικόνες, δεύτερες προεικόνες ή να βρίσκει συγκρούσεις. Προσέξτε ότι για τον υπολογισμό προεικόνων για παράδειγμα κανείς μπορεί πάντα να επιλέγει στοιχεία του X τυχαία και να ελέγχει αν

$h(x) = y$. Η πιθανότητα επιτυχίας μιας τέτοιας στρατηγικής όμως είναι αμελητέα αν το $|h^{-1}(y)|/|X|$ είναι αρκετά μικρό. Φυσικά κάτι τέτοιο ισχύει για κάθε καλά σχεδιασμένη συνάρτηση κατακερματισμού.