Κρυπτογραφία

Πιστοποίηση μηνύματος (Message authentication) -
Πιστοποίηση ταυτότητας αποστολέα (Entity authentication)

Επιπρόσθετες απαιτήσεις στην κρυπτογραφία

- Ακεραιότητα δεδομένων (data integrity):Επιβεβαίωση ότι το μήνυμα που αποκρυπτογραφείται είναι το αυθεντικό και δεν έχει παραποιηθεί κατά τη μετάδοσή του
- Πιστοποίηση ταυτότητας ή αυθεντικοποίηση (identification ή entity authentication) : Επιβεβαίωση ότι ο αποστολέας είναι πράγματι αυτός που ισχυρίζεται

Συναρτήσεις κατακερματισμού (hash functions)

- Δέχονται είσοδο οσουδήποτε μήκους και παράγουν έξοδο σταθερού μήκους (ίσου ή μικρότερου του μεγέθους της εισόδου) – (συμπίεση)
- Η τιμή της συνάρτησης για οποιαδήποτε είσοδό της υπολογίζεται εύκολα. Το αντίστροφο όμως δεν ισχύει: για οποιοδήποτε y, δεν μπορεί να βρεθεί x ώστε h(x)=y. (preimage resistance). Είναι λοιπόν συναρτήσεις μιας κατεύθυνσης.
- Για οποιοδήποτε δοθέν M, είναι υπολογιστικά δύσκολη η εύρεση M' με την ιδιότητα $h(M) = h(M')$ (2nd-preimage resistance)
- Δεν μπορούν να υπολογιστούν δύο διαφορετικές είσοδοι M, M' που να δίνουν την ίδια έξοδο, δηλαδή h(M)=h(M') (collision resistance)
- Χωρίζονται σε δύο κατηγορίες – σε αυτές που υπεισέρχεται και κάποιο κλειδί στον υπολογισμό τους (keyed hash functions) και σε αυτές που όχι (unkeyed hash functions)
- Αν M το μήνυμα και H η συνάρτηση κατακερματισμού, τότε το H(M) αποκαλείται αποτύπωμα (fingerprint) ή σύνοψη του μηνύματος (Message Digest – MD)

Μία απλή συνάρτηση κατακερματισμού

- Bitwise-XOR (ένα μπλοκ bits μήκους nm γίνεται ένα μικρότερο μπλοκ μήκους n, με χρήση XOR όπως

	bit 1	bit 2	• • •	bit *n*
block 1	b_{11}	b_{21}		b_{n1}
block 2	b_{12}	b_{22}		b_{n2}
	•	•	•	•
	•	•	•	•
	•	•	•	•
block *m*	b_{1m}	b_{2m}		b_{nm}
hash code	C_1	C_2		C_n

- Όχι καλή: Δεν ικανοποιεί το 2nd preimage resistance.

Κάθε συνάρτηση μιας κατεύθυνσης είναι συνάρτηση κατακερματισμού??

- ΌΧΙ. Ας θεωρήσουμε για παράδειγμα την

$$h(x) = x^2 \bmod n,$$

χωρίς να είναι γνωστή η παραγοντοποίηση n=pq.

- Δεν ικανοποιεί το 2nd preimage resistance: για δοθέν x, προφανώς το –x δίνει την ίδια έξοδο.

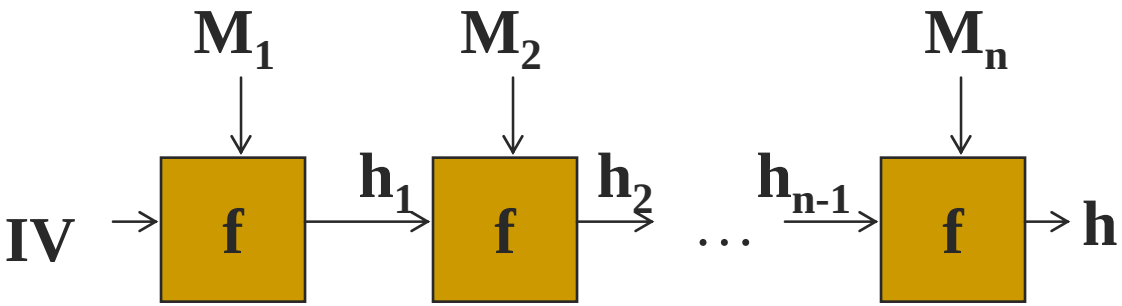
Κατηγορίες αλγορίθμων με συναρτήσεις κατακερματισμού – MDCs και MACs

- MDC (Modification Detection Codes – Κώδικας ανίχνευσης τροποποίησης)
  - Συναρτήσεις κατακερματισμού στις οποίες δεν υπεισέρχεται κλειδί. Είναι μίας κατεύθυνσης και Ικανοποιούν το preimage resistance
- MAC (Message Authentication Codes – Κώδικας αυθεντικοποίησης μηνύματος)
  - Συναρτήσεις κατακερματισμού στις οποίες υπεισέρχεται επιπρόσθετα και ένα μυστικό κλειδί. Είναι μίας κατεύθυνσης και ικανοποιούν και αυτές το preimage resistance.

- Και στις δύο περιπτώσεις η συνάρτηση κατακερματισμού είναι δημοσίως γνωστή.

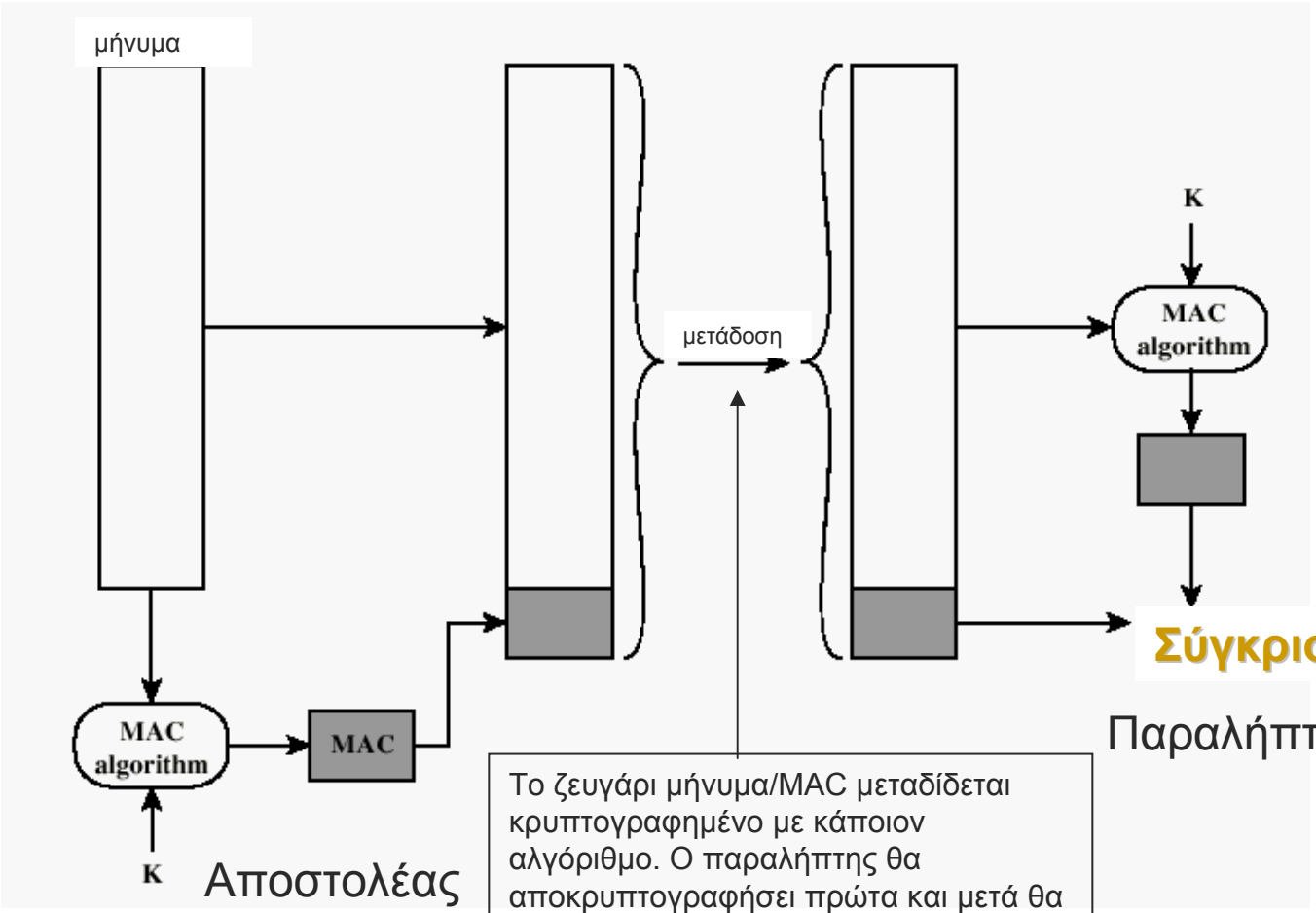
Κατασκευή Συναρτήσεων Κατακερματισμού μίας κατεύθυνσης

- Συνήθως κατασκευάζονται από συναρτήσεις των οποίων το πλήθος εξόδων είναι μικρότερο από το πλήθος εισόδων (για να επιτευχθεί η συμπίεση)
- Η δομή αυτή είναι ανάλογη με ένα αλυσιδωτό block cipher (Chained Block Cipher - CBC)
  - Παράγει μία τιμή κατακερματισμού για κάθε μπλοκ σταθερού μήκους, με βάση το περιεχόμενό του αλλά και την τιμή κατακερματιμού του προηγούμενου μπλοκ



- Ο Rabin πρότεινε κάθε συνάρτηση f να είναι ο αλγόριθμος του DES (τα M_i θα έχουν το ρόλο του κλειδιού)

Πιστοποίηση μηνύματος με χρήση MAC



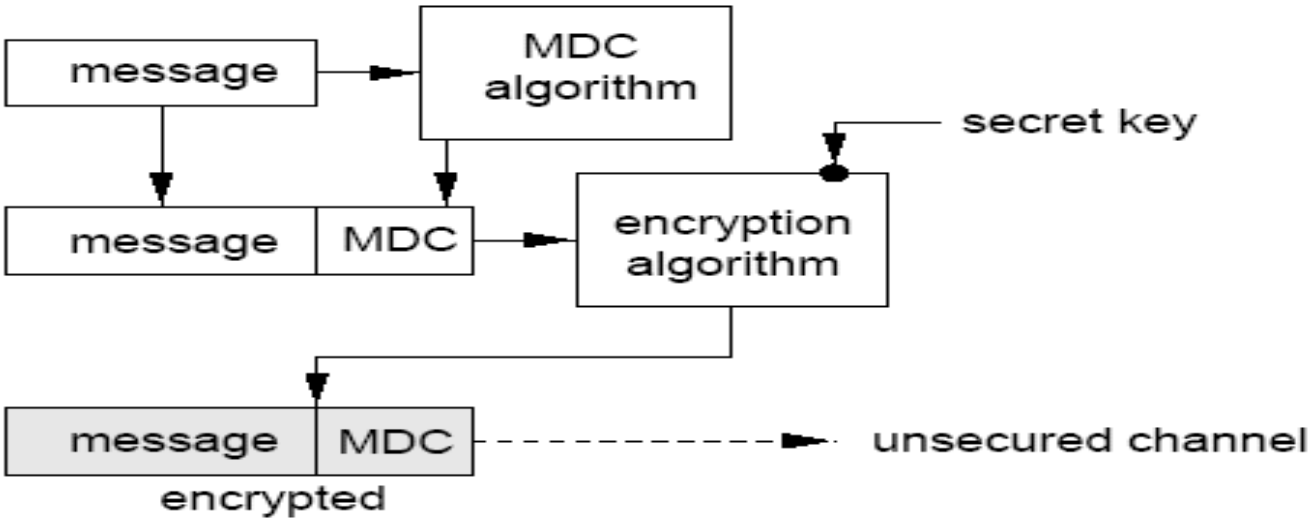
Ιδιότητες MAC

- Το MAC επιτυγχάνει τα παρακάτω:
  - Το μήνυμα είναι γνήσιο
    - Αν κάποιος «εισβολέας» τροποποιήσει το μήνυμα, θα φανεί στη σύγκριση που κάνει ο δέκτης
  - Η πηγή του μηνύματος είναι γνήσια
    - Κανείς άλλος δεν έχει το ίδιο κλειδί για να δημιουργήσει το ίδιο MAC
- Οποιοσδήποτε μπλοκ κρυπτογραφικός αλγόριθμος μπορεί να χρησιμοποιηθεί για τη δημιουργία του MAC (όπως ήδη είδαμε στους τρόπους λειτουργίας των μπλοκ αλγορίθμων, και συγκεκριμένα στο τρόπο λειτουργίας CBC).

Ασφάλεια του MAC

- Η ασφάλεια του MAC καθορίζεται από το μέγεθος του κλειδιού (θεωρούμε ότι ο επιτιθέμενος γνωρίζει τόσο το κρυπτογραφημένο μήνυμα M αλλά τον MAC κώδικα $H_k(M)$). Αν ανακαλύψει το κλειδί k, μπορεί να στείλει δικά του μηνύματα λανθασμένα προσποιούμενος ότι είναι ο πραγματικός αποστολέας (ο παραλήπτης δεν θα το καταλαβαίνει).
- Προσέξτε όμως ότι ακόμα κι αν ο επιτιθέμενος ανακαλύψει ένα κλειδί k τέτοιο ώστε αν εφαρμόσει το H_k στο M να πάρει το $H_k(M)$, ίσως το κλειδί να μην είναι το πραγματικό (μια που ενδεχομένως δύο διαφορετικά κλειδιά μπορούν να δίνουν το ίδιο αποτέλεσμα, αφού το μήκος του κατακερματισμένου μηνύματος είναι μικρότερο από το μήκος του αρχικού μηνύματος).

Πιστοποίηση μηνύματος με χρήση συνάρτησης κατακερματισμού (MDC)



- Μία συνάρτηση κατακερματισμού εφαρμόζεται στο μήνυμα – το αποτέλεσμα προστίθεται στο τέλος του μηνύματος και όλο το νέο μπλοκ κρυπτογραφείται και μεταδίδεται
- Ο δέκτης αποκρυπτογραφεί και κάνει σύγκριση ανάλογη με την περίπτωση του MAC

Αλγόριθμοι κατακερματισμού

	SHA-1	MD5 (MD4+)	RIPEMD-160
Μήκος εξόδου	160 bits	128 bits	160 bits
Πλήθος βημάτων	80 (4 γύροι των 20)	64 (4 γύροι των 16)	160 (5 ζευγάρια γύρων των 16)
Μέγιστο μέγεθος μηνύματος	2^{64}-1 bits	απεριόριστο	απεριόριστο

Ποιο το “σωστό” μήκος του κατακερματισμένου μηνύματος??

- Το μέγεθος της εξόδου μιας συνάρτησης κατακερματισμού είναι κρίσιμο: αφενός για λόγους απόδοσης του συστήματος θέλουμε να είναι μικρό, αφετέρου όμως μικρές τιμές ενδεχομένως να διευκολύνουν έναν επιτιθέμενο να βρει ένα ψεύτικο μήνυμα που να κατακερματίζεται στην ίδια τιμή.
- Για το μέγεθος της εξόδου της συνάρτησης κατακερματισμού, καλό είναι να θυμόμαστε πάντα το μαθηματικό παράδοξο των γενεθλίων

Παράδοξο των γενεθλίων (Birthday paradox)

- Έστω ότι 23 άτομα βρίσκονται σε μία δεξίωση. Αν αναλογιστούμε την πιθανότητα δύο από αυτά να έχουν γενέθλια την ίδια μέρα (όχι απαραίτητα του ίδιου έτους), τότε με μία βιαστική εκτίμηση θα μπορούσε να πει κάποιος ότι η πιθανότητα αυτή είναι μικρή. Θα δούμε όμως ότι η πιθανότητα αυτή είναι μεγαλύτερη από 50%!!

Παράδοξο των γενεθλίων - απόδειξη

- Εάν έχουμε n άτομα και p_n είναι η πιθανότητα τουλάχιστον δύο άτομα να έχουν ημερομηνία την ίδια μέρα, τότε η πιθανότητα όλοι να έχουν διαφορετικές μέρες γενέθλια ισούται με $p'_n=1-p_n$.
- Έστω ότι τα άτομα εισέρχονται ένα-ένα. Η πιθανότητα του πρώτου ατόμου να μην έχει ίδια μέρα γενέθλια με κανέναν από όσους είναι ήδη στο δωμάτιο είναι προφανώς 1 ή, ισοδύναμα, 365/365.
- Η αντίστοιχη πιθανότητα για το δεύτερο άτομο που μπαίνει στην αίθουσα είναι προφανώς 364/365=(365-1)/365.
- Παρόμοια, η αντίστοιχη πιθανότητα για το τρίτο άτομο είναι 363/365=(365-2)/365.
- Με την ίδια λογική, όταν και το n-ιοστό άτομο μπει στην αίθουσα, η πιθανότητα κανείς να μην έχει ίδια μέρα γενέθλια με κανέναν ισούται με

$$p'_n = \frac{365}{365} \cdot \frac{(365-1)}{365} \cdot \frac{(365-2)}{365} \cdot \ldots \cdot \frac{(365-n+1)}{365} = \frac{365!}{(365-n)!365^n}$$

Παράδοξο των γενεθλίων – απόδειξη (συνέχεια)

- Άρα, η πιθανότητα να υπάρχει τουλάχιστον ένα ζευγάρι ανθρώπων που να έχουν γενέθλια την ίδια μέρα, είναι

$$p_n = 1 - \frac{365!}{(365 - n)!365^n}$$

Για n=23, η πιθανότητα p_n γίνεται 0,507 – δηλαδή πάνω από 50%!!!

Κρυπτογραφική αξία: Το παράδοξο των γενεθλίων είναι καλό να το έχει πάντα στο μυαλό του όποιος σχεδιάζει κρυπτογραφική συνάρτηση κατακερματισμού. Μπορεί δηλαδή διαισθητικά να πιστεύουμε ότι είναι αρκετά απίθανο δύο διαφορετικές είσοδοι σε μία συνάρτηση κατακερματισμού να δίνουν ίδια έξοδο, αλλά παρόλα αυτά η πιθανότητα τελικά να είναι πολύ μεγαλύτερη από ό,τι δείχνει φαινομενικά.

Μη δυνατότητα αποποίησης (κάποιας ενέργειας) με συναρτήσεις κατακερματισμού

- Έστω ότι κάποιος βρίσκει δύο μηνύματα M1, M2, τέτοια ώστε H(M1)=H(M2).
- Στέλνει το M1 σε κάποιον άλλο
- Στη συνέχεια, ισχυρίζεται ότι έστειλε το M2! Ο παραλήπτης δεν μπορεί να το αποδείξει.

- Άρα, με τις συναρτήσεις κατακερματισμού, αντιμετωπίζει κανείς και το πρόβλημα της αποποίησης (εκτός της ακεραιότητας των δεδομένων που ήδη αναφέραμε).

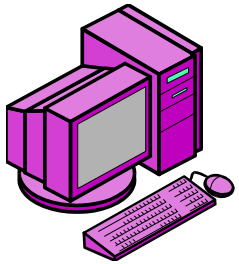
Πιστοποίηση ταυτότητας (entity authentication)

- Ανάγκη επιβεβαίωσης ότι αυτός που με τον οποίο μιλάει κανείς είναι πραγματικά αυτός που ισχυρίζεται
- Η διαφορά με τις διάφορες μεθόδους επιβεβαίωσης της ταυτότητας του συνομιλητή που αναφέρθηκαν παραπάνω στα πλαίσια της πιστοποίησης μηνύματος (π.χ. MAC), είναι ότι μιλάμε πια για πιστοποίηση ταυτότητας σε πραγματικό χρόνο

Πώς επιτυγχάνεται ισχυρή πιστοποίηση ταυτότητας ενός χρήστη με χρήση συνάρτησης κατακερματισμού??

Ο C έχει ένα password το οποίο γνωρίζει ο V

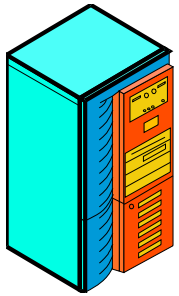
1.
Ο V δημιουργεί το μήνυμα – Πρόκληση (Challenge Message)



Claimant
(C)

Πρόκληση

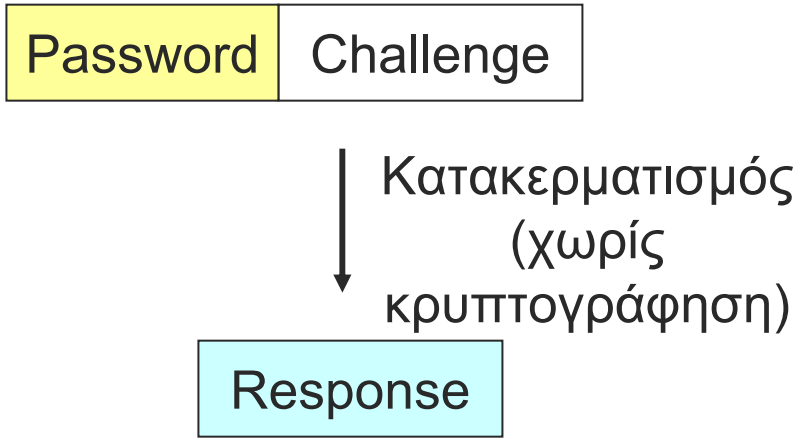
2.
Ο V στέλνει την πρόκληση



Verifier
(V)

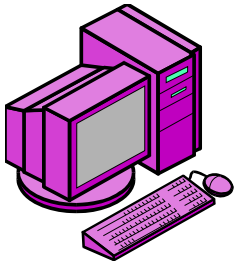
Πώς επιτυγχάνεται ισχυρή πιστοποίηση ταυτότητας ενός χρήστη με χρήση συνάρτησης κατακερματισμού?? (II)

3.
Ο C παράγει το μήνυμα-απάντηση με τον εξής τρόπο:
(a) Επισυνάπτει το password στο μήνυμα-πρόκληση
(b) Κατακερματίζει το αποτέλεσμα (χωρίς κρυπτογράφηση)
(c) Το κατακερματισμένο μήνυμα είναι η απάντηση (Response Message)



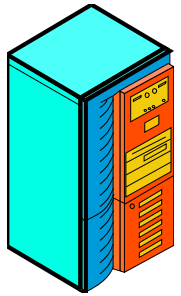
Πώς επιτυγχάνεται ισχυρή πιστοποίηση ταυτότητας ενός χρήστη με χρήση συνάρτησης κατακερματισμού?? (III)

4.
Ο C στέλνει την απάντηση, χωρίς να την κρυπτογραφήσει



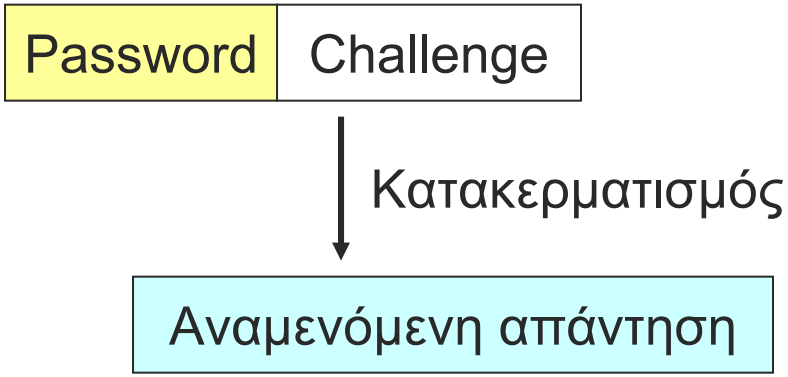
Claimant
(C)

Απάντηση (challenge response)



Verifier
(V)

Πώς επιτυγχάνεται ισχυρή πιστοποίηση ταυτότητας ενός χρήστη με χρήση συνάρτησης κατακερματισμού?? (IV)



5.
Ο V επισυνάπτει το password στο μήνυμα που έστειλε και κατακερματίζει το αποτέλεσμα. Αυτό που προκύπτει είναι η αναμενόμενη απάντηση από τον C

Πώς επιτυγχάνεται ισχυρή πιστοποίηση ταυτότητας ενός χρήστη με χρήση συνάρτησης κατακερματισμού?? (VI)

Μεταδιδόμενη απάντηση **=?** Αναμενόμενη απάντηση

6.
Αν η ληφθείσα απάντηση ταυτίζεται με την αναμενόμενη, τότε
Ο V συμπεραίνει ότι ο C ξέρει το σωστό password.

Στην παραπάνω ανάλυση δεν χρησιμοποιήθηκε καθόλου κρυπτογράφηση, παρά μόνο κατακερματισμός. Ωστόσο, απαραίτητη προϋπόθεση είναι να γνωρίζει ο Verifier το password (κάτι που δεν χρειάζεται στα πρωτόκολλα μηδενικής γνώσης που περιγράφονται παρακάτω).

Άλλες τεχνικές πιστοποίησης ταυτότητας

- Με χρήση αλγορίθμων Δημοσίου Κλειδιού (κρυπτογράφηση του κάθε μηνύματος με το ιδιωτικό κλειδί του χρήστη – βλέπε διαφάνειες κεφαλαίου 1).
- Οι RSA, El Gamal, Rabin είναι οι βασικοί αλγόριθμοι που χρησιμοποιούνται για πιστοποίηση ταυτότητας του χρήστη.
- Υπάρχει μια ειδική τεχνική, η οποία βασίζεται σε αλγορίθμους δημοσίου κλειδιού, αλλά έχει διαφορετική φιλοσοφία – είναι τα λεγόμενα πρωτόκολλα μηδενικής γνώσης (Zero-knowledge protocols) (π.χ. Fiat-Shamir) – αναπτύσσεται στο εργαστήριο (δεν θα μελετηθεί εδώ)

Ψηφιακή υπογραφή (Digital Signature)

- Υπογραφή (signature) $s=S_I(m)$ για ένα μήνυμα m ενός χρήστη I
  - ο I υπογράφει, δηλαδή *υπολογίζει το* s, συνάρτηση τόσο του m όσο και του ίδιου του χρήστη I
  - Για δοθέντα s,I, και m, *ο οποιοσδήποτε* μπορεί να ελέγξει ότι $s=S_I(m)$
  - ***Η συνάρτηση υπολογισμού του s πρέπει να είναι κατάλληλη ώστε να μη μπορεί κάποιος άλλος να δημιουργήσει το ίδιο s με τον I***

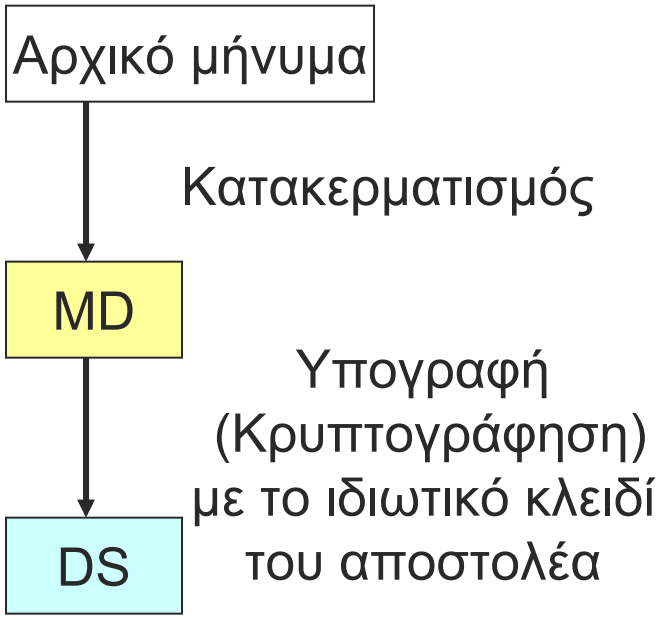
Ψηφιακή υπογραφή με χρήση αλγορίθμου Δημοσίου Κλειδιού

Για τη δημιουργία της ψηφιακής υπογραφής:

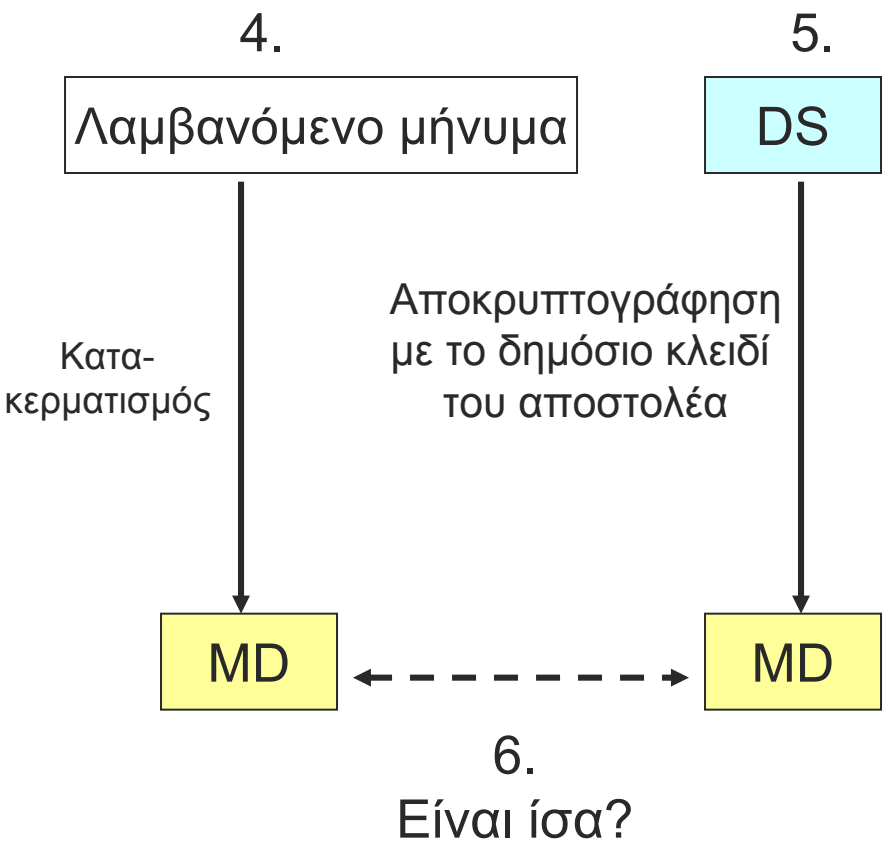
1. Κατακερματισμός του αρχικού μηνύματος, έτσι ώστε να προκύψει ένα «αποτύπωμα» (MD) του μηνύματος

2. Το MD κρυπτογραφείται, με το ιδιωτικό κλειδί του αποστολέα. Προκύπτει έτσι η ψηφιακή υπογραφή (DS)

3. Η υπογραφή επισυνάπτεται στο αρχικό μήνυμα, κρυπτογραφείται το αποτέλεσμα (στην τυπική περίπτωση με συμμετρικό κλειδί) και μεταδίδεται



Έλεγχος ψηφιακής υπογραφής, με αλγόριθμο Δημοσίου Κλειδιού



Για να ελέγξει ο δέκτης την υπογραφή:

4. Κατακερματίζει το αρχικό μήνυμα με την ίδια συνάρτηση κατακερματισμού. Αυτό οδηγεί στον υπολογισμό του MD.

5. Αποκρυπτογραφεί την υπογραφή με το Δημόσιο Κλειδί του αποστολέα. Αυτό επίσης οδηγεί στον υπολογισμό του MD

6. Αν συμπίπτουν, η ταυτότητα του αποστολέα επιβεβαιώνεται

Αλγόριθμοι ψηφιακών υπογραφών

- Όλες οι γνωστές μέθοδοι Δημοσίου Κλειδιού μπορούν να χρησιμοποιηθούν για την ψηφιακή υπογραφή
  - RSA (χρησιμοποιείται ευρέως – βλέπε εργαστηριακές σημειώσεις (για όποιον ενδιαφέρεται))
  - Rabin (βασίζεται στον αλγόριθμο κρυπτογράφησης Rabin που μελετήθηκε στο Κεφάλαιο 4. Χρησιμοποιεί μία συνάρτηση πλεονασμού. Δεν θα μελετηθεί εδώ. Όποιος ενδιαφέρεται, μπορεί να ανατρέξει στο Κεφάλαιο 11 του Handbook of Applied Cryptography).
  - El Gamal
  - DSA (παραλλαγή του El Gamal)
- Άλλοι αλγόριθμοι ψηφιακής υπογραφής
  - GOST
  - Ong-Schnorr-Shamir
  - ESIGN

Αλγόριθμος ψηφιακής υπογραφής El Gamal

- Κάθε χρήστης επιλέγει:
  - Έναν πολύ μεγάλο πρώτο αριθμό p
  - Έναν ακέραιο g που είναι **γεννήτορας mod p** (αυτό σημαίνει ότι $g^k \bmod p \neq 1 \bmod p$ για όλα τα $1 \leq k \leq p-2$.
  - Έναν τυχαίο ακέραιο $1 \leq a \leq p-2$
  - Υπολογίζει τον $y=g^a \bmod p$
  - Ιδιωτικό κλειδί: a
  - Δημόσιο κλειδί: p, g, y

Αλγόριθμος ψηφιακής υπογραφής El Gamal (2)

- Ο χρήστης υπογράφει το μήνυμα m ως εξής:
  - Κατακερματίζει το μήνυμα με κάποιον γνωστό αλγόριθμο κατακερματισμού – υπολογίζει έτσι το h(m)
  - Επιλέγει τυχαίο $1 \leq k \leq p-2$, τέτοιο ώστε gcd(k,p-1)=1.
  - Υπολογίζει το $r = g^k \bmod p$
  - Υπολογίζει το $k^{-1} \bmod (p-1)$ - δηλαδή, τον ακέραιο αριθμό τέτοιον ώστε $k \cdot k^{-1} \equiv 1 \bmod (p-1)$
  - Υπολογίζει το $s = k^{-1}(h(m) - ar) \bmod (p-1)$
  - Η υπογραφή είναι το ζευγάρι (r,s)

Αλγόριθμος ψηφιακής υπογραφής El Gamal (3)

- Ο παραλήπτης ελέγχει την υπογραφή ως εξής:
  - Βρίσκει το h(m) (εφόσον έχει λάβει το μήνυμα m)
  - Υπολογίζει το $u_1=y^r \cdot r^s \bmod p$
  - Υπολογίζει το $u_2 = g^{h(m)} \bmod p$
  - Η υπογραφή είναι έγκυρη μόνο αν $u_1=u_2$

Αλγόριθμος ψηφιακής υπογραφής El Gamal (3)

- Απόδειξη της ορθότητας του αλγόριθμου υπογραφής El Gamal
  - Εάν η υπογραφή είναι σωστή, τότε ισχύει

    $s \equiv k^{-1}(h(m) - ar) \bmod (p-1)$.
  - Πολλαπλασιάζοντας με το k και τα δύο μέλη, έχουμε

    $h(m)=ks + ar \bmod (p-1)$.
  - Άρα, $g^{h(m)} \equiv g^{ks+ar} \equiv r^s y^r \bmod (p-1)$

Αλγόριθμος ψηφιακής υπογραφής El Gamal - Παράδειγμα

- Για λόγους απλότητας θεωρούμε h(m)=m.
- Έστω το μήνυμα m=5
- Δημόσιο κλειδί: p=11, g=6, y=3
- Ιδιωτικό κλειδί: a=2 (προσέξτε ότι ισχύει $y \equiv g^a \bmod p$)
- Ο χρήστης επιλέγει τυχαίο k=7 και υπολογίζει το $r = g^k \bmod p = 8$.
- Υπολογίζει το $7^{-1} \bmod 10 = 3$ (πράγματι, $7 \cdot 3 \equiv 1 \bmod 10$. Το 3 μπορεί να το βρει με δοκιμές, ελέγχοντας όλους τους αριθμούς από 1 μέχρι p-2=9). Άρα, $k^{-1}=3$.
- Υπολογίζει το $s = k^{-1}(h(m) – ar) \bmod (p-1) =$

  $= 3(5-16) \bmod 10 \equiv 3 \cdot (-11) \bmod 10 \equiv 3 \cdot 9 \bmod 10 = 7$.
- Υπογραφή: (r,s)=(8,7)

Αλγόριθμος ψηφιακής υπογραφής El Gamal – Παράδειγμα (συνέχεια)

- Ο παραλήπτης λαμβάνει το μήνυμα m=5 και την υπογραφή (r,s)=(8,7). Ξέρει επίσης το δημόσιο κλειδί (p=11, g=6, y=3) του αποστολέα.
  - Βρίσκει το h(m)=h(5)=5
  - Υπολογίζει το $u_1=y^r \cdot r^s \bmod p =$ $3^8 \cdot 8^7 \bmod 11 = 5 \cdot 2 \bmod 11 = 10.$
  - Υπολογίζει το $u_2 = g^{h(m)} \bmod p =$ $=6^5 \bmod 11 = 10$
  - Η υπογραφή είναι έγκυρη γιατί $u_1=u_2$

Αλγόριθμος DSA (Digital Signature Algorithm)*

- Προτάθηκε από τον NIST το 1991
- Απαιτεί τη χρήση συνάρτησης κατακερματισμού (SHA-1)
- Στηρίχτηκε στον αλγόριθμο ElGamal

* Όσες διαφάνειες από αυτές που ακολουθούν έχουν αστερίσκο *, είναι εκτός ύλης

Περιγραφή DSA *

- Παράμετροι συστήματος
  - Επιλογή ενός πρώτου q των 160-bit
  - Επιλογή ενός πρώτου p των 1024-bit, έτσι ώστε: $q \mid p-1$
  - Επιλογή $g \in Z_p^*$ και υπολογισμός $a = g^{(p-1)/q} \bmod p$
    - *Αν a=1, η διαδικασία επαναλαμβάνεται με διαφορετικό g*
- Κλειδιά
  - *Επιλογή τυχαίου μυστικού κλειδιού x* $(1 \leq x \leq q-1)$
  - Υπολογισμός Δημοσίου Κλειδιού $y = a^x \bmod p$ *(όλα τα υπόλοιπα νούμερα είναι επίσης δημόσια)*

Υπογραφή στον DSA*

- Για να υπογραφεί ένα μήνυμα *m*
  - Κατακερματισμός του $m \rightarrow h(m)$ $(1 \leq h(m) \leq q-1)$ *(με τον αλγόριθμο κατακερματισμού SHA-1)*
  - Δημιουργία τυχαίου (και μυστικού) k $(1 \leq k \leq q-1)$
  - Υπολογισμός $r = (a^k \bmod p) \bmod q$
  - Υπολογισμός $k^{-1} \bmod q$
  - Υπολογισμός $s = k^{-1}\{h(m) + xr\} \bmod q$
  - Η υπογραφή στο *m* είναι (r,s)

DSA – Επιβεβαίωση υπογραφής*

- Για την επιβεβαίωση των *(r,s)*
  - Έλεγχος του ότι $1 \leq r \leq q-1$ και $1 \leq s \leq q-1$
  - Υπολογισμός $w = s^{-1} \bmod q$
  - *Υπολογισμός h(m)*
  - Υπολογισμός $u_1 = wh(m) \bmod q$
  - Υπολογισμός $u_2 = rw \bmod q$
  - Αποδοχή της υπογραφής αν
    - $(a^{u1} y^{u2} \bmod p) \bmod q = r$

Ερμηνεία του αλγορίθμου*

- Εάν το (r,s) είναι σωστή και έγκυρη υπογραφή, τότε ισχύει

  $h(m) \equiv -xr + ks \bmod q$
- Πολλαπλασιάζοντας με το w και τα δύο μέλη, προκύπτει
- $wh(m) + xrw \equiv k \bmod q \Rightarrow$

  $\Rightarrow u_1 + xu_2 \equiv k \bmod q \Rightarrow a^{u1 + xu2} \equiv a^k \bmod q \Rightarrow$

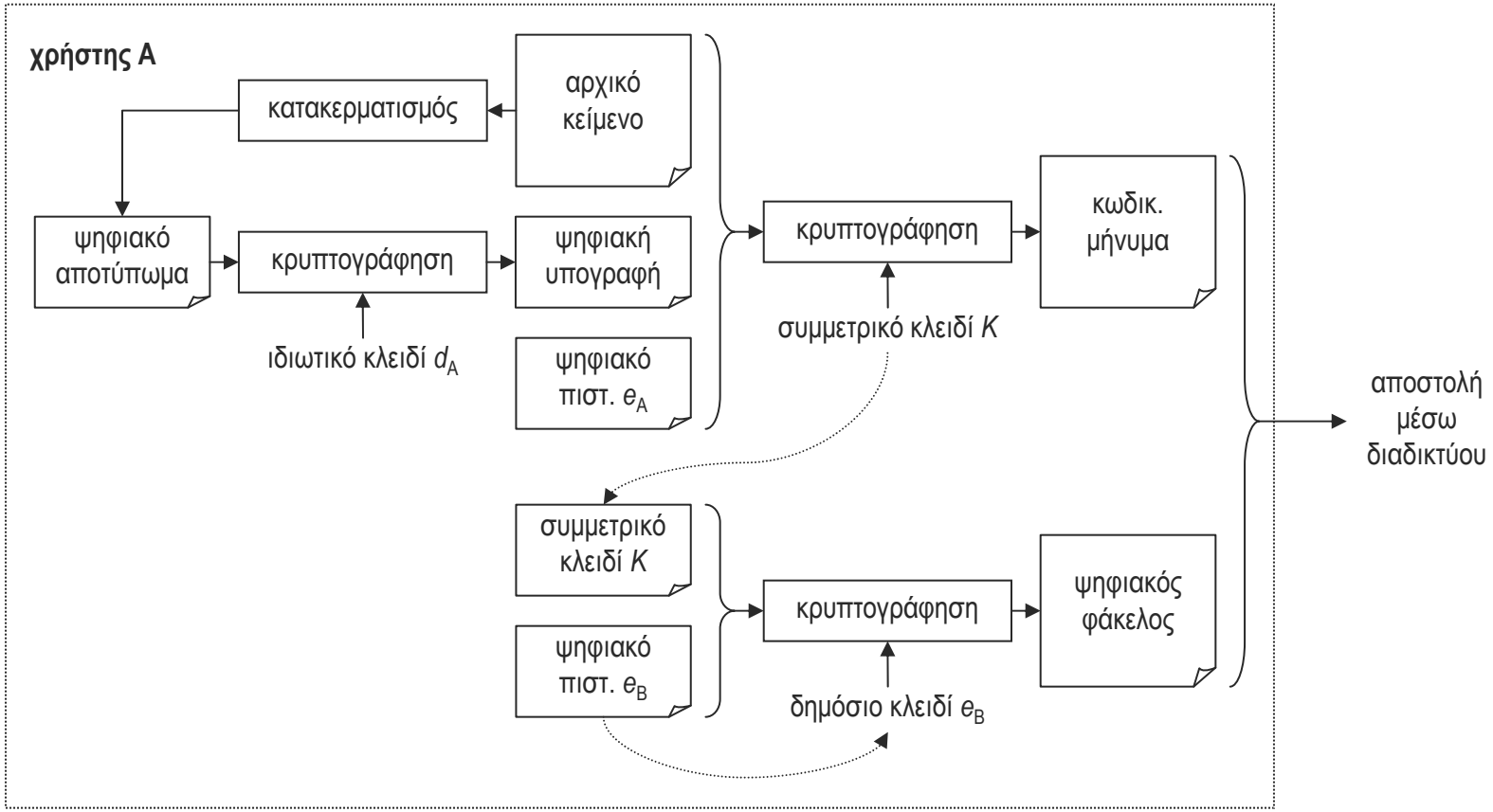
  $\Rightarrow \boxed{(a^{u1}\, y^{u2} \bmod p) \bmod q \equiv (a^k \bmod p) \bmod q}$

Ασφάλεια του DSA*

- Έγκειται στα:
  - Υπολογισμός λογαρίθμων στο *GF(p)*
  - Υπολογισμός λογαρίθμων σε κυκλική υπο-ομάδα τάξης *q*
    - Αλγόριθμοι για αυτό χρειάζονται χρόνο ανάλογο του $q^{1/2}$ *(κεφάλαιο 3 – Handbook of Applied Cryptography)*
    - Στην πράξη επιλέγεται $q \approx 2^{160}$ και $p \approx 2^{1024}$

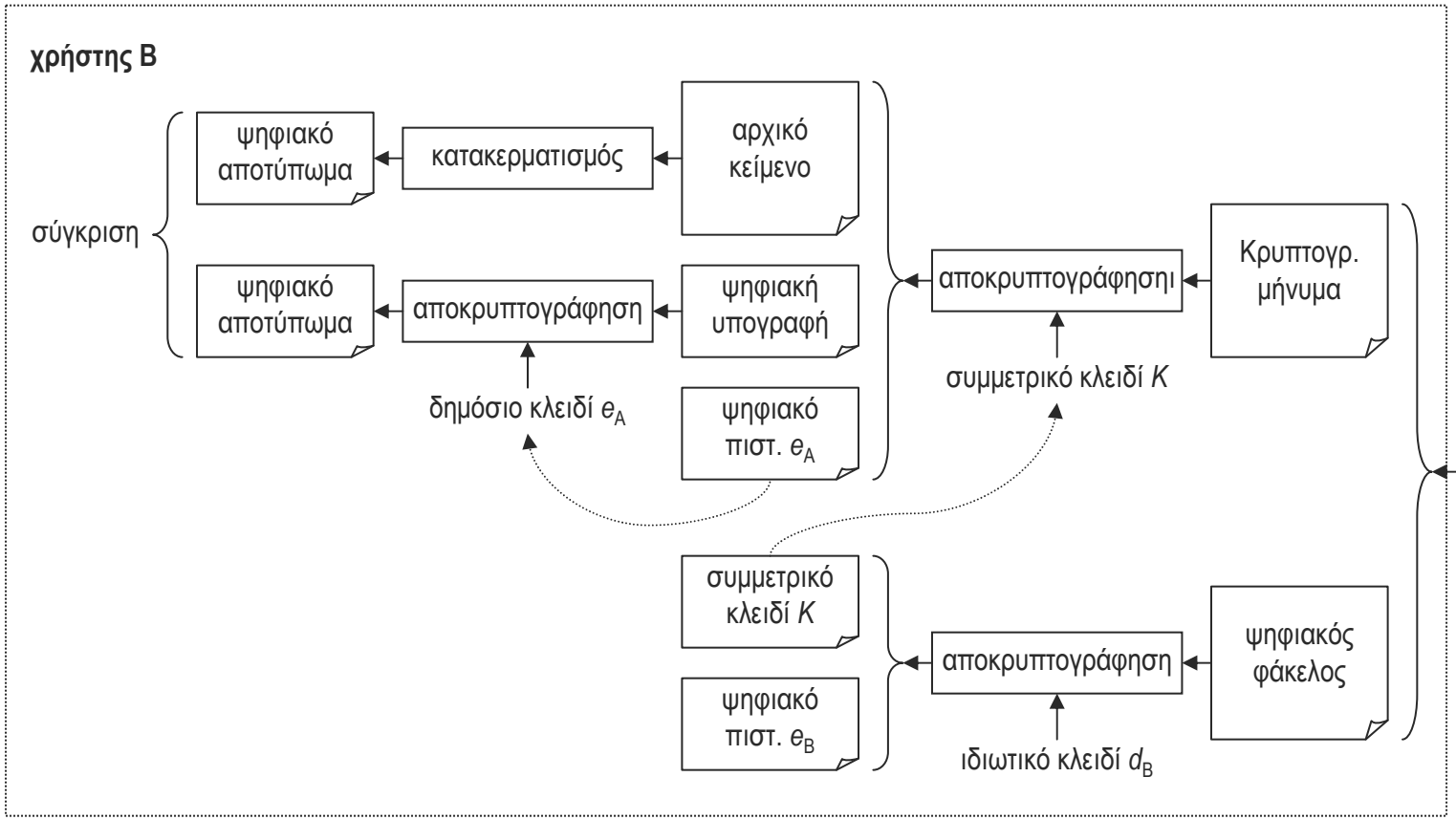
Συνδυασμός Μεθοδολογιών - Κρυπτογράφηση

Εξασφάλιση εμπιστευτικότητας - ακεραιότητας - αυθεντικότητας



Συνδυασμός Μεθοδολογιών - Αποκρυπτογράφηση

Εξασφάλιση εμπιστευτικότητας - ακεραιότητας - αυθεντικότητας



Χαρακτηριστικά συστήματα με την προηγούμενη δομή

- PGP (Pretty Good Privacy) (www.pgp.com)
  - Ο κάθε χρήστης πιστοποιεί το δημόσιο κλειδί του άλλου χρήστη, δηλαδή ο κάθε χρήστης μπορεί να γίνει μία αρχή πιστοποίησης.
- X509 (Πρότυπο για το Internet, για υποδομή Δημοσίου κλειδιού)
  - Δεν καθορίζονται οι συμμετρικοί αλγόριθμοι κρυπτογράφησης που θα χρησιμοποιηθούν για την κρυπτογράφηση των δεδομένων