

# Κρυπτογραφία

MAC - Γνησιότητα/Ακεραιότητα μηνύματος

Πέτρος Ποτίκας

Εθνικό Μετσόβιο Πολυτεχνείο  
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

# Περιεχόμενα

- 1 Message Authentication Code (MAC)
- 2 Ψευδοτυχαίες συναρτήσεις ως κώδικες γνησιότητας
- 3 HMAC
- 4 Ιδιωτικότητα και γνησιότητα

- ▶ Κρυπτογράφηση κρύβει μήνυμα από αντίπαλο

- ▶ Κρυπτογράφηση κρύβει μήνυμα από αντίπαλο
- ▶ Φτάνει για ασφαλή επικοινωνία;

- ▶ Κρυπτογράφηση κρύβει μήνυμα από αντίπαλο
- ▶ Φτάνει για ασφαλή επικοινωνία;
- ▶ Γνησιότητα/ακεραιότητα μηνύματος (*message integrity/authentication*)

- ▶ Κρυπτογράφηση κρύβει μήνυμα από αντίπαλο
- ▶ Φτάνει για ασφαλή επικοινωνία;
- ▶ Γνησιότητα/ακεραιότητα μηνύματος (*message integrity/authentication*)
- ▶ Παράδειγμα: Η εταιρεία Β παίρνει παραγγελία από την εταιρεία Α να φτιάξει 1000 οθόνες  
Ερωτήματα:

- ▶ Κρυπτογράφηση κρύβει μήνυμα από αντίπαλο
  - ▶ Φτάνει για ασφαλή επικοινωνία;
  - ▶ *Γνησιότητα/ακεραιότητα μηνύματος (message integrity/authentication)*
  - ▶ Παράδειγμα: Η εταιρεία B παίρνει παραγγελία από την εταιρεία A να φτιάξει 1000 οθόνες
- Ερωτήματα:

1. Το έστειλε πράγματι η A;

- ▶ Κρυπτογράφηση κρύβει μήνυμα από αντίπαλο
- ▶ Φτάνει για ασφαλή επικοινωνία;
- ▶ *Γνησιότητα/ακεραιότητα μηνύματος (message integrity/authentication)*
- ▶ Παράδειγμα: Η εταιρεία B παίρνει παραγγελία από την εταιρεία A να φτιάξει 1000 οθόνες

Ερωτήματα:

1. Το έστειλε πράγματι η A;
2. Είναι το 1000 σωστό;



# Message Authentication Code (MAC)

- ▶ Δύο παίκτες θέλουν να επικοινωνήσουν ανταλλάσσοντας ‘ακέραια μηνύματα’

# Message Authentication Code (MAC)

- ▶ Δύο παίκτες θέλουν να επικοινωνήσουν ανταλλάσσοντας ‘ακέραια μηνύματα’
- ▶ Ανταλλάσσουν κοινό μυστικό κλειδί  $k$

# Message Authentication Code (MAC)

- ▶ Δύο παίκτες θέλουν να επικοινωνήσουν ανταλλάσσοντας ‘ακέραια μηνύματα’
- ▶ Ανταλλάσσουν κοινό μυστικό κλειδί  $k$
- ▶ Όταν ένας παίκτης θέλει να στείλει ένα μήνυμα  $m$ , υπολογίζει μια ετικέτα  $t$  (tag) με βάση το μήνυμα και το κοινό τους κλειδί, με έναν αλγόριθμο παραγωγής ετικέτας  $Mac$

# Message Authentication Code (MAC)

- ▶ Δύο παίκτες θέλουν να επικοινωνήσουν ανταλλάσσοντας ‘ακέραια μηνύματα’
- ▶ Ανταλλάσσουν κοινό μυστικό κλειδί  $k$
- ▶ Όταν ένας παίκτης θέλει να στείλει ένα μήνυμα  $m$ , υπολογίζει μια ετικέτα  $t$  (tag) με βάση το μήνυμα και το κοινό τους κλειδί, με έναν αλγόριθμο παραγωγής ετικέτας  $Mac$
- ▶ Στέλνει το μήνυμα μαζί με την ετικέτα  $(m, t)$

# Message Authentication Code (MAC)

- ▶ Δύο παίκτες θέλουν να επικοινωνήσουν ανταλλάσσοντας ‘ακέραια μηνύματα’
- ▶ Ανταλλάσσουν κοινό μυστικό κλειδί  $k$
- ▶ Όταν ένας παίκτης θέλει να στείλει ένα μήνυμα  $m$ , υπολογίζει μια ετικέτα  $t$  (tag) με βάση το μήνυμα και το κοινό τους κλειδί, με έναν αλγόριθμο παραγωγής ετικέτας  $Mac$
- ▶ Στέλνει το μήνυμα μαζί με την ετικέτα  $(m, t)$
- ▶ Ο άλλος παίκτης λαμβάνει το  $(m, t)$  και επιβεβαιώνει τη γνησιότητα του μηνύματος (αν το  $t$  είναι έγκυρο για το  $m$  με βάση το κοινό κλειδί που έχει, με έναν αλγόριθμο επαλήθευσης  $Vrfy$ )

# Message Authentication Code (MAC)

## Ορισμός

Ένας κώδικας γνησιότητας μηνύματος (*Message Authentication Code, MAC*) είναι μια πλειάδα αλγορίθμων ( $Gen, Mac, Vrfy$ ), τέτοιων ώστε:

1. Ο αλγόριθμος παραγωγής κλειδιού  $Gen$  παίρνει είσοδο την παράμετρο ασφαλείας  $1^n$  και επιστρέφει ένα κλειδί  $k$ , με  $|k| \geq n$
2. Ο αλγόριθμος παραγωγής ετικέτας  $Mac$  παίρνει σαν είσοδο ένα κλειδί  $k$  και ένα μήνυμα  $m \in \{0, 1\}^*$  και επιστρέφει μια ετικέτα  $t \leftarrow Mac_k(m)$
3. Ο αλγόριθμος επαλήθευσης  $Vrfy$  παίρνει σαν είσοδο ένα  $k$ , ένα  $m$  και μια  $t$  και επιστρέφει 1, αν η ετικέτα είναι έγκυρη, αλλιώς 0 (ντετερμινιστικός αλγόριθμος)

Για κάθε  $n$ , κάθε  $k$  που παράγεται από τον  $Gen$  και κάθε  $m \in \{0, 1\}^*$ , ισχύει  $Vrfy_k(m, Mac_k(m)) = 1$

Αν το MAC ορίζεται μόνο για μηνύματα μήκους  $l(n)$ , τότε λέγεται *σταθερού μήκους*

# Ασφάλεια κώδικα γνησιότητας μηνύματος

Διαισθητικά: κανένας αντίπαλος δεν μπορεί να φτιάξει μια έγκυρη ετικέτα για ένα “νέο” μήνυμα που δεν έχει γνησιότητα ακόμα

Ο αντίπαλος βλέπει τα  $(m, t)$  που ανταλλάσσονται και μπορεί να τα αλλάζει

## Πείραμα γνησιότητας μηνύματος $\text{Mac-forge}_{\mathcal{A}, \Pi}(n)$

1. Ένα τυχαίο κλειδί παράγεται από τον  $\text{Gen}(1^n)$
2. Ο αντίπαλος  $\mathcal{A}$  παίρνει σαν είσοδο το  $1^n$  και πρόσβαση σε ένα μαντείο  $\text{Mac}_k(\cdot)$ . Δίνει σαν έξοδο ένα  $(m, t)$  και  $Q$  το σύνολο των ερωτήσεων που κάνει στο μαντείο
3. Η έξοδος του πειράματος είναι 1, ανν (1)  $\text{Vrfy}_k(m, t) = 1$  και (2)  $m \notin Q$

# Ασφάλεια κώδικα γνησιότητας μηνύματος

## Ορισμός

Ένας κώδικας γνησιότητας μηνύματος  $\Pi = (Gen, Mac, Vrfy)$  είναι *υπαρξιακά μη-παραχαράξιμος σε μια προσαρμοζόμενη επίθεση επιλεγμένου μηνύματος* (*existentially unforgeable under an adaptive chosen-message attack*) ή *ασφαλής* αν για κάθε PPT αντίπαλο  $\mathcal{A}$  υπάρχει μια αμελητέα συνάρτηση  $negl$  τέτοια ώστε:

$$Pr[\text{Mac-forge}_{\mathcal{A}, \Pi}(n) = 1] \leq \text{negl}(n)$$



## Επίθεση επανάληψης

Ένας αντίπαλος μπορεί να στείλει ένα μήνυμα και την ετικέτα ενός προηγούμενου έγκυρου μηνύματος

### Παράδειγμα:

Η Alice στέλνει μήνυμα στην τράπεζά της να δώσουν 1000 € στον Bob

## Επίθεση επανάληψης

Ένας αντίπαλος μπορεί να στείλει ένα μήνυμα και την ετικέτα ενός προηγούμενου έγκυρου μηνύματος

### Παράδειγμα:

Η Alice στέλνει μήνυμα στην τράπεζά της να δώσουν 1000 € στον Bob  
Ο Bob δεν μπορεί να το κάνει 10.000 €, αλλά μπορεί να το στείλει 10 φορές στην τράπεζα!

## Επίθεση επανάληψης

Ένας αντίπαλος μπορεί να στείλει ένα μήνυμα και την ετικέτα ενός προηγούμενου έγκυρου μηνύματος

### Παράδειγμα:

Η Alice στέλνει μήνυμα στην τράπεζά της να δώσουν 1000 € στον Bob  
Ο Bob δεν μπορεί να το κάνει 10.000 €, αλλά μπορεί να το στείλει 10 φορές στην τράπεζα!

MAC δεν έχουν την ικανότητα να αποφύγουν τέτοιες επιθέσεις, γιατί δε δίνουν σημασιολογία

Είναι θέμα εφαρμογής, τι σημαίνει η επανάληψη μηνύματος

## Επίθεση επανάληψης

Ένας αντίπαλος μπορεί να στείλει ένα μήνυμα και την ετικέτα ενός προηγούμενου έγκυρου μηνύματος

### Παράδειγμα:

Η Alice στέλνει μήνυμα στην τράπεζά της να δώσουν 1000 € στον Bob  
Ο Bob δεν μπορεί να το κάνει 10.000 €, αλλά μπορεί να το στείλει 10 φορές στην τράπεζα!

MAC δεν έχουν την ικανότητα να αποφύγουν τέτοιες επιθέσεις, γιατί δε δίνουν σημασιολογία

Είναι θέμα εφαρμογής, τι σημαίνει η επανάληψη μηνύματος

Άμυνα:

1. Ένας μοναδικός αριθμός μαζί με το μήνυμα
2. Timestamp, η ώρα προστίθεται στο μήνυμα

## Επίθεση επανάληψης - Μοναδικός αριθμός

- ▶ Σε κάθε μήνυμα  $m$  αντιστοιχεί ένας μοναδικός αριθμός  $i$ , η γνησιότητα υπολογίζεται στο  $i||m$

## Επίθεση επανάληψης - Μοναδικός αριθμός

- ▶ Σε κάθε μήνυμα  $m$  αντιστοιχεί ένας μοναδικός αριθμός  $i$ , η γνησιότητα υπολογίζεται στο  $i||m$
- ▶ Ο αποστολέας πάντα αναθέτει ένα μοναδικό αριθμό σε κάθε μήνυμα, ο παραλήπτης φυλάει αυτούς τους αριθμούς

## Επίθεση επανάληψης - Μοναδικός αριθμός

- ▶ Σε κάθε μήνυμα  $m$  αντιστοιχεί ένας μοναδικός αριθμός  $i$ , η γνησιότητα υπολογίζεται στο  $i||m$
- ▶ Ο αποστολέας πάντα αναθέτει ένα μοναδικό αριθμό σε κάθε μήνυμα, ο παραλήπτης φυλάει αυτούς τους αριθμούς
- ▶ Επιτυχημένη επίθεση στο  $m$ : δημιουργία έγκυρης ετικέτας σε ένα νέο μήνυμα  $i' || m$ , όπου  $i'$  είναι φρέσκο

## Επίθεση επανάληψης - Μοναδικός αριθμός

- ▶ Σε κάθε μήνυμα  $m$  αντιστοιχεί ένας μοναδικός αριθμός  $i$ , η γνησιότητα υπολογίζεται στο  $i||m$
- ▶ Ο αποστολέας πάντα αναθέτει ένα μοναδικό αριθμό σε κάθε μήνυμα, ο παραλήπτης φυλάει αυτούς τους αριθμούς
- ▶ Επιτυχημένη επίθεση στο  $m$ : δημιουργία έγκυρης ετικέτας σε ένα νέο μήνυμα  $i' || m$ , όπου  $i'$  είναι φρέσκο
- ▶ Μειονέκτημα: φύλαξη όλων των αριθμών από τον παραλήπτη



## Επίθεση επανάληψης - Timestamp

- ▶ Η τρέχουσα ώρα (στο κοντινότερο millisecond) προστίθεται στο μήνυμα, ο παραλήπτης αποδέχεται αν είναι εντός ενός περιθωρίου
- ▶ Μειονέκτημα: τα συμβαλλόμενα μέρη πρέπει να έχουν συγχρονισμένα ρολόγια και μπορεί ο αντίπαλος να προλάβει να ξαναστείλει κάτι (πόσο στενό είναι το χρονικό περιθώριο)

# Ψευδοτυχαίες συναρτήσεις ως Κώδικες Γνησιότητας

- ▶ Οι ψευδοτυχαίες συναρτήσεις είναι κατάλληλες για κώδικες γνησιότητας

## Ψευδοτυχαίες συναρτήσεις ως Κώδικες Γνησιότητας

- ▶ Οι ψευδοτυχαίες συναρτήσεις είναι κατάλληλες για κώδικες γνησιότητας
- ▶ Αν  $t$  παράγεται από την εφαρμογή μιας ψευδοτυχαίας συνάρτησης σε ένα μήνυμα  $m$ , τότε η παραχάραξη απαιτεί να μαντέψει ο αντίπαλος την τιμή μιας ψευδοτυχαίας συνάρτησης σε ένα “νέο” μήνυμα

## Ψευδοτυχαίες συναρτήσεις ως Κώδικες Γνησιότητας

- ▶ Οι ψευδοτυχαίες συναρτήσεις είναι κατάλληλες για κώδικες γνησιότητας
- ▶ Αν  $t$  παράγεται από την εφαρμογή μιας ψευδοτυχαίας συνάρτησης σε ένα μήνυμα  $m$ , τότε η παραχάραξη απαιτεί να μαντέψει ο αντίπαλος την τιμή μιας ψευδοτυχαίας συνάρτησης σε ένα “νέο” μήνυμα
- ▶ Πιθανότητα να μαντέψει σωστά:  $2^{-n}$ , όταν μήκος εξόδου συνάρτησης είναι  $n$

## Ψευδοτυχαίες συναρτήσεις ως Κώδικες Γνησιότητας

- ▶ Οι ψευδοτυχαίες συναρτήσεις είναι κατάλληλες για κώδικες γνησιότητας
- ▶ Αν  $t$  παράγεται από την εφαρμογή μιας ψευδοτυχαίας συνάρτησης σε ένα μήνυμα  $m$ , τότε η παραχάραξη απαιτεί να μαντέψει ο αντίπαλος την τιμή μιας ψευδοτυχαίας συνάρτησης σε ένα “νέο” μήνυμα
- ▶ Πιθανότητα να μαντέψει σωστά:  $2^{-n}$ , όταν μήκος εξόδου συνάρτησης είναι  $n$

### Κατασκευή

Έστω  $F$  μια ψευδοτυχαία συνάρτηση. Ορίζουμε έναν καθορισμένου μήκους κώδικα γνησιότητας για μηνύματα μήκους  $n$  ως:

- ▶ *Gen*: με είσοδο  $1^n$ , επίλεξε  $k \leftarrow \{0, 1\}^n$
- ▶ *Mac*: με είσοδο  $k, m \in \{0, 1\}^n$ , δώσε στην έξοδο  $t := F_k(m)$  (αν  $|m| \neq |k|$  μη δίνεις αποτέλεσμα)
- ▶ *Vrfy*: με είσοδο  $k, m, t$ , δώσε αποτέλεσμα 1, αν  $t = F_k(m)$  (αν  $|m| \neq |k|$ , δώσε 0)

# Ασφάλεια κατασκευής

## Θεώρημα

*Αν η  $F$  είναι μια ψευδοτυχαία συνάρτηση, τότε η παραπάνω κατασκευή είναι ένας καθορισμένου μήκους κώδικας γνησιότητας για μηνύματα μήκους  $n$  που είναι υπαρξιακά μη-παραχαράξιμος σε μια προσαρμοζόμενη επίθεση επιλεγμένου μηνύματος*

## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$

## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$
- ▶ Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο



## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$
  - ▶ Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
  - ▶ Ιδέες:
1. Πάρε το XOR όλων των block και εφάρμοσε την  $MAC$ ,  
 $t := MAC'_k(\oplus_i(m_i))$

## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$
- ▶ Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
- ▶ Ιδέες:

1. Πάρε το XOR όλων των block και εφάρμοσε την  $MAC$ ,

$$t := MAC'_k(\oplus_i(m_i))$$

Ο αντίπαλος μπορεί να αλλάξει το αρχικό μήνυμα ώστε το XOR να μείνει ίδιο

## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$
- ▶ Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
- ▶ Ιδέες:
  1. Πάρε το XOR όλων των block και εφάρμοσε την  $MAC$ ,  
 $t := MAC'_k(\oplus_i(m_i))$   
Ο αντίπαλος μπορεί να αλλάξει το αρχικό μήνυμα ώστε το XOR να μείνει ίδιο
  2. Πάρε κάθε block ξεχωριστά,  $t := \langle MAC'_k(m_1), \dots, MAC'_k(m_d) \rangle$

## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$
- ▶ Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
- ▶ Ιδέες:

1. Πάρε το XOR όλων των block και εφάρμοσε την  $MAC$ ,

$$t := MAC'_k(\oplus_i(m_i))$$

Ο αντίπαλος μπορεί να αλλάξει το αρχικό μήνυμα ώστε το XOR να μείνει ίδιο

2. Πάρε κάθε block ξεχωριστά,  $t := \langle MAC'_k(m_1), \dots, MAC'_k(m_d) \rangle$

Αν αλλάξεις σειρά στα blocks;

## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$
- ▶ Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
- ▶ Ιδέες:

1. Πάρε το XOR όλων των block και εφάρμοσε την  $MAC$ ,

$$t := MAC'_k(\oplus_i(m_i))$$

Ο αντίπαλος μπορεί να αλλάξει το αρχικό μήνυμα ώστε το XOR να μείνει ίδιο

2. Πάρε κάθε block ξεχωριστά,  $t := \langle MAC'_k(m_1), \dots, MAC'_k(m_d) \rangle$

Αν αλλάξεις σειρά στα blocks;

3. Πάρε block ξεχωριστά μαζί με έναν αριθμό

$$t := \langle MAC'_k(1||m_1), \dots, MAC'_k(d||m_d) \rangle$$

## Επέκταση σε μηνύματα μεταβλητού μήκους

- ▶ Έστω  $\Pi = (Gen', Mac', Vrfy')$  ένας κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$
- ▶ Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
- ▶ Ιδέες:

1. Πάρε το XOR όλων των block και εφάρμοσε την  $MAC$ ,

$$t := MAC'_k(\oplus_i(m_i))$$

Ο αντίπαλος μπορεί να αλλάξει το αρχικό μήνυμα ώστε το XOR να μείνει ίδιο

2. Πάρε κάθε block ξεχωριστά,  $t := \langle MAC'_k(m_1), \dots, MAC'_k(m_d) \rangle$

Αν αλλάξεις σειρά στα blocks;

3. Πάρε block ξεχωριστά μαζί με έναν αριθμό

$$t := \langle MAC'_k(1||m_1), \dots, MAC'_k(d||m_d) \rangle$$

Μπορεί να πετάξει block από τέλος ή να συνδυάσει προηγούμενα μηνύματα

## Επέκταση σε μηνύματα μεταβλητού μήκους

Επιπλέον πληροφορία, ένα τυχαίο “αναγνωριστικό μηνύματος” σε κάθε block και το μήκος μηνύματος

### Κατασκευή

Έστω  $\Pi' = (Gen', Mac', Vrfy')$  ένας MAC καθορισμένου μήκους για μηνύματα μήκους  $n$ . Ορίζουμε ένα MAC ως εξής:

- ▶  $Gen'$ : ίδιο με το  $Gen'$
- ▶  $Mac'$ : με είσοδο  $k \in \{0, 1\}^n$ ,  $m \in \{0, 1\}^*$  μήκους  $l < 2^{n/4}$ , ανάλυσέ το σε  $d$  blocks,  $m_1, \dots, m_d$ , καθένα μήκους  $n/4$  (συμπλήρωσε με μηδενικά αν χρειάζεται). Διάλεξε αναγνωριστικό  $r \leftarrow \{0, 1\}^{n/4}$   
Υπολόγισε  $t_i := Mac'_k(r || l || i || m_i)$ ,  $1 \leq i \leq d$ , και δώσε έξοδο  $t := \langle r, t_1, \dots, t_d \rangle$
- ▶  $Vrfy'$ : με είσοδο  $k \in \{0, 1\}^n$ ,  $m \in \{0, 1\}^*$  μήκους  $l < 2^{n/4}$  και  $t := \langle r, t_1, \dots, t_d \rangle$ , ανάλυσε το  $m$  σε  $d'$  blocks, καθένα μήκους  $n/4$  (συμπλήρωσε με μηδενικά αν χρειάζεται). Έξοδος 1 ανν (1)  $d = d'$  και (2)  $Vrfy'_k(r || l || i || m_i) = t_i$ ,  $1 \leq i \leq d$

# Ασφάλεια κατασκευής

## Θεώρημα

*Αν  $\Pi'$  είναι ασφαλής κώδικας γνησιότητας καθορισμένου μήκους για μηνύματα μήκους  $n$ , τότε η παραπάνω κατασκευή είναι υπαρξιακά μη-παραχαράξιμη σε μια προσαρμοζόμενη επίθεση επιλεγμένου μηνύματος για μηνύματα μεταβλητού μήκους*

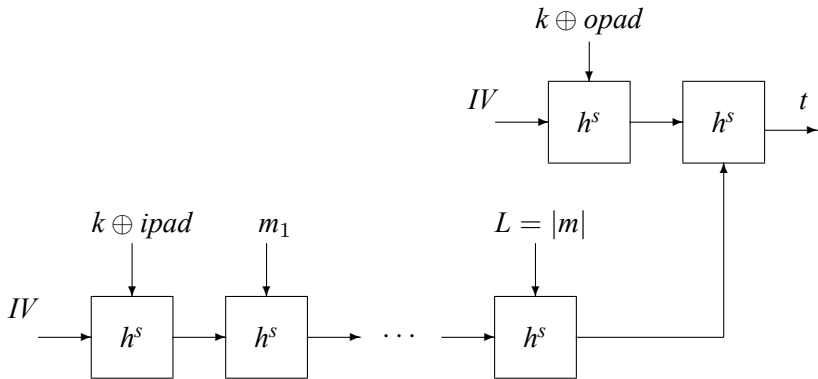


# Hashed MAC (HMAC)

## Κατασκευή HMAC

Έστω  $(Gen, h)$  μια συνάρτηση σύνοψης καθορισμένου μήκους που αντιστέκεται σε συγκρούσεις και  $(Gen, H)$  το αποτέλεσμα του Merkle-Damgård μετασχηματισμού. Έστω  $ipad, opad$  σταθερές μήκους  $n$ . Ο HMAC ορίζεται ως εξής:

- ▶  $Gen$ : με είσοδο  $1^n$ , τρέξε  $Gen(1^n)$  για να πάρεις το κλειδί  $s$ . Επίλεξε  $k \leftarrow \{0, 1\}^n$  και δώσε στην έξοδο το κλειδί  $(s, k)$
- ▶  $Mac$ : με είσοδο  $(s, k)$ , ένα  $m \in \{0, 1\}^*$  δώσε  $t := H^s((k \oplus opad) || H^s(k \oplus ipad) || m)$
- ▶  $Vrfy$ : με είσοδο  $(s, k)$ ,  $m \in \{0, 1\}^*$  και μια ετικέτα  $t$ , δώσε 1 αν  $t \stackrel{?}{=} Mac_{s,k}(m)$



Σχήμα : HMAC

- ▶ Η ασφάλεια του HMAC μπορεί να βασιστεί στην ασθενέστερη υπόθεση της ασφάλειας του δεύτερου ορίσματος

# ipad και opad

- ▶ Η ασφάλεια του HMAC μπορεί να βασιστεί στην ασθενέστερη υπόθεση της ασφάλειας του δεύτερου ορίσματος
- ▶ Αρκεί ένα κλειδί

# HMAC στην πράξη

- ▶ Το HMAC είναι πρότυπο και χρησιμοποιείται στην πράξη (SSL, SSH)

# HMAC στην πράξη

- ▶ Το HMAC είναι πρότυπο και χρησιμοποιείται στην πράξη (SSL, SSH)
- ▶ Είναι αποδοτικό

# HMAC στην πράξη

- ▶ Το HMAC είναι πρότυπο και χρησιμοποιείται στην πράξη (SSL, SSH)
- ▶ Είναι αποδοτικό
- ▶ Εύκολα υλοποιήσιμο

# HMAC στην πράξη

- ▶ Το HMAC είναι πρότυπο και χρησιμοποιείται στην πράξη (SSL, SSH)
- ▶ Είναι αποδοτικό
- ▶ Εύκολα υλοποιήσιμο
- ▶ Έχει απόδειξη ότι είναι ασφαλές (βασισμένο σε υποθέσεις που πιστεύουμε ότι ισχύουν για συναρτήσεις σύνοψης)



# Ιδιωτικότητα και γνησιότητα

- ▶ Μπορούμε να συνδυάσουμε ένα σύστημα κρυπτογράφησης με ένα σύστημα γνησιότητας μηνύματος;
- ▶ Στις περισσότερες online εργασίες, ιδίως τραπεζικές συναλλαγές χρειάζονται και τα δύο

# Ιδιωτικότητα και γνησιότητα

- ▶ Μπορούμε να συνδυάσουμε ένα σύστημα κρυπτογράφησης με ένα σύστημα γνησιότητας μηνύματος;
- ▶ Στις περισσότερες online εργασίες, ιδίως τραπεζικές συναλλαγές χρειάζονται και τα δύο
- ▶ Κρυπτογράφηση αρχείων στο δίσκο: κρυπτογράφηση μόνο είναι αρκετή;

# Ιδιωτικότητα και γνησιότητα

- ▶ Μπορούμε να συνδυάσουμε ένα σύστημα κρυπτογράφησης με ένα σύστημα γνησιότητας μηνύματος;
- ▶ Στις περισσότερες online εργασίες, ιδίως τραπεζικές συναλλαγές χρειάζονται και τα δύο
- ▶ Κρυπτογράφηση αρχείων στο δίσκο: κρυπτογράφηση μόνο είναι αρκετή; Αν κάποιος αλλάξει στοιχεία, τότε π.χ. μια εταιρεία θα βγάλει λάθος εκθέσεις

# Ιδιωτικότητα και γνησιότητα

- ▶ Μπορούμε να συνδυάσουμε ένα σύστημα κρυπτογράφησης με ένα σύστημα γνησιότητας μηνύματος;
- ▶ Στις περισσότερες online εργασίες, ιδίως τραπεζικές συναλλαγές χρειάζονται και τα δύο
- ▶ Κρυπτογράφηση αρχείων στο δίσκο: κρυπτογράφηση μόνο είναι αρκετή; Αν κάποιος αλλάξει στοιχεία, τότε π.χ. μια εταιρεία θα βγάλει λάθος εκθέσεις
- ▶ Συμβουλή: πάντα συνδυασμός γνησιότητα μηνύματος με ιδιωτικότητα/κρυπτογράφηση

# Ιδιωτικότητα και γνησιότητα

- ▶ Μπορούμε να συνδυάσουμε ένα σύστημα κρυπτογράφησης με ένα σύστημα γνησιότητας μηνύματος;
- ▶ Στις περισσότερες online εργασίες, ιδίως τραπεζικές συναλλαγές χρειάζονται και τα δύο
- ▶ Κρυπτογράφηση αρχείων στο δίσκο: κρυπτογράφηση μόνο είναι αρκετή; Αν κάποιος αλλάξει στοιχεία, τότε π.χ. μια εταιρεία θα βγάλει λάθος εκθέσεις
- ▶ Συμβουλή: πάντα συνδυασμός γνησιότητα μηνύματος με ιδιωτικότητα/κρυπτογράφηση (εξαιρέση: λίγους πόρους)

# Ιδιωτικότητα και γνησιότητα

- ▶ Μπορούμε να συνδυάσουμε ένα σύστημα κρυπτογράφησης με ένα σύστημα γνησιότητας μηνύματος;
- ▶ Στις περισσότερες online εργασίες, ιδίως τραπεζικές συναλλαγές χρειάζονται και τα δύο
- ▶ Κρυπτογράφηση αρχείων στο δίσκο: κρυπτογράφηση μόνο είναι αρκετή; Αν κάποιος αλλάξει στοιχεία, τότε π.χ. μια εταιρεία θα βγάλει λάθος εκθέσεις
- ▶ Συμβουλή: πάντα συνδυασμός γνησιότητα μηνύματος με ιδιωτικότητα/κρυπτογράφηση (εξάιρεση: λίγους πόρους)
- ▶ Οποιοσδήποτε συνδυασμός δεν είναι σωστός

# Ιδιωτικότητα και γνησιότητα

# Ιδιωτικότητα και γνησιότητα

Έστω  $k_1$  το κλειδί κρυπτογράφησης,  $k_2$  γνησιότητας (ΠΑΝΤΑ ανεξάρτητα)



# Ιδιωτικότητα και γνησιότητα

Έστω  $k_1$  το κλειδί κρυπτογράφησης,  $k_2$  γνησιότητας (ΠΑΝΤΑ ανεξάρτητα)

## 1. Κρυπτογράφηση-και-γνησιότητα

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(m)$$

# Ιδιωτικότητα και γνησιότητα

Έστω  $k_1$  το κλειδί κρυπτογράφησης,  $k_2$  γνησιότητας (ΠΑΝΤΑ ανεξάρτητα)

## 1. Κρυπτογράφηση-και-γνησιότητα

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(m)$$

## 2. Γνησιότητα-μετά-κρυπτογράφηση

$$t \leftarrow Mac_{k_2}(m) \text{ και } c \leftarrow Enc_{k_1}(m||t)$$

# Ιδιωτικότητα και γνησιότητα

Έστω  $k_1$  το κλειδί κρυπτογράφησης,  $k_2$  γνησιότητας (ΠΑΝΤΑ ανεξάρτητα)

## 1. Κρυπτογράφηση-και-γνησιότητα

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(m)$$

## 2. Γνησιότητα-μετά-κρυπτογράφηση

$$t \leftarrow Mac_{k_2}(m) \text{ και } c \leftarrow Enc_{k_1}(m||t)$$

## 3. Κρυπτογράφηση-μετά-γνησιότητα

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(c)$$

# Ανάλυση ασφαλείας

- ▶ Θέλουμε όλοι οι συνδυασμοί σχημάτων κρυπτογράφησης και MACs να είναι ασφαλείς

# Ανάλυση ασφαλείας

- ▶ Θέλουμε όλοι οι συνδυασμοί σχημάτων κρυπτογράφησης και MACs να είναι ασφαλείς
- ▶ Σημείωση: Μπορεί να υπάρχει ένα σχήμα από καθένα από αυτά ώστε να είναι ασφαλές

# Ανάλυση ασφαλείας

- ▶ Θέλουμε όλοι οι συνδυασμοί σχημάτων κρυπτογράφησης και MACs να είναι ασφαλείς
- ▶ Σημείωση: Μπορεί να υπάρχει ένα σχήμα από καθένα από αυτά ώστε να είναι ασφαλές
- ▶ Θέλουμε όλα, ώστε να ελαχιστοποιήσουμε λάθη στην υλοποίηση (αντικατάσταση με νεότερες εκδόσεις ή αλλαγή των standards)

# Ανάλυση ασφάλειας

Ορισμός “ασφαλούς συνδυασμού”

Έστω  $\Pi_E = (Gen_E, Enc, Dec)$  σχήμα κρυπτογράφησης,

$\Pi_M = (Gen_M, Mac, Vrfy)$  σχήμα γνησιότητας μηνύματος.

Ένα σχήμα μετάδοσης μηνύματος  $\Pi' = (Gen', EncMac', Dec')$  που παράγεται από τα  $\Pi_E, \Pi_M$  είναι μια πλειάδα αλγορίθμων που κάνουν τα εξής:

- ▶ Ο  $Gen'$  τρέχει  $Gen_E(1^n), Gen_M(1^n)$  και παράγει τα κλειδιά  $k_1, k_2$ , αντίστοιχα
- ▶ Ο  $EncMac'$  με είσοδο τα  $k_1, k_2$  και  $m$  δίνει ένα  $c$  τρέχοντας κάποιο συνδυασμό των  $Enc_{k_1}, Mac_{k_2}$
- ▶ Ο  $Dec'$  παίρνει είσοδο τα  $k_1, k_2$  και ένα  $c$  και εφαρμόζει κάποιο συνδυασμό των  $Dec_{k_1}, Vrfy_{k_2}$  και δίνει έξοδο είτε το  $m$  είτε  $\perp$  για σφάλμα

# Ανάλυση ασφάλειας

Για την ορθότητα του σχήματος απαιτούμε για κάθε  $n$ , κάθε κλειδιά  $k_1, k_2$  που παράγονται από την  $Gen'$  και κάθε  $m \in \{0, 1\}^*$  να έχουμε

$$Dec'_{k_1, k_2}(EncMac'_{k_1, k_2}(m)) = m$$



# Ανάλυση ασφάλειας

Για την ορθότητα του σχήματος απαιτούμε για κάθε  $n$ , κάθε κλειδιά  $k_1, k_2$  που παράγονται από την  $Gen'$  και κάθε  $m \in \{0, 1\}^*$  να έχουμε

$$Dec'_{k_1, k_2}(EncMac'_{k_1, k_2}(m)) = m$$

Το  $\Pi'$  ικανοποιεί συντακτικά το σχήμα κρυπτογράφησης ιδιωτικού κλειδιού, γιατί έχουμε κρυπτογράφηση όπου επιπλέον ζητάμε γνησιότητα

# Ανάλυση ασφάλειας

Για τον ορισμό της ασφάλειας του  $\Pi'$  θα ορίσουμε ξεχωριστές έννοιες ιδιωτικότητας και γνησιότητας

Η έννοια της ιδιωτικότητας που θεωρούμε είναι ότι το  $\Pi'$  η CCA-secure (αναβάθμιση από CPA-secure)

Για την γνησιότητα θεωρούμε ότι είναι υπαρξιακά μη-παραχαράξιμο σε μια προσαρμοζόμενη επίθεση επιλεγμένου μηνύματος

Το  $\Pi'$  δεν ικανοποιεί το συντακτικό ορισμό ενός σχήματος γνησιότητας μηνύματος, άρα θα πρέπει να κάνουμε αλλαγές

# Ανάλυση ασφάλειας

Για σχήμα  $\Pi'$ , αντίπαλο  $\mathcal{A}$  και παράμετρο ασφαλείας  $n$  ορίζουμε:

## Πείραμα ασφαλούς μετάδοσης μηνύματος $\text{Auth}_{\mathcal{A}, \Pi'}(n)$

1. Ένα κλειδί  $k = (k_1, k_2)$  παράγεται από τον  $\text{Gen}'(1^n)$
2. Ο αντίπαλος  $\mathcal{A}$  παίρνει σαν είσοδο το  $1^n$  και πρόσβαση σε ένα μαντείο  $\text{EncMac}'_k$ . Δίνει έξοδο ένα  $c$ .  $\mathcal{Q}$ : το σύνολο των ερωτήσεων που κάνει στο μαντείο
3. Έστω  $m := \text{Dec}'_k(c)$ . Η έξοδος είναι 1, ανν (1)  $m \neq \perp$  και (2)  $m \notin \mathcal{Q}$

# Ανάλυση ασφάλειας

Για σχήμα  $\Pi'$ , αντίπαλο  $\mathcal{A}$  και παράμετρο ασφαλείας  $n$  ορίζουμε:

## Πείραμα ασφαλούς μετάδοσης μηνύματος $\text{Auth}_{\mathcal{A},\Pi'}(n)$

1. Ένα κλειδί  $k = (k_1, k_2)$  παράγεται από τον  $\text{Gen}'(1^n)$
2. Ο αντίπαλος  $\mathcal{A}$  παίρνει σαν είσοδο το  $1^n$  και πρόσβαση σε ένα μαντείο  $\text{EncMac}'_k$ . Δίνει έξοδο ένα  $c$ .  $\mathcal{Q}$ : το σύνολο των ερωτήσεων που κάνει στο μαντείο
3. Έστω  $m := \text{Dec}'_k(c)$ . Η έξοδος είναι 1, ανν (1)  $m \neq \perp$  και (2)  $m \notin \mathcal{Q}$

## Ορισμός

Ένα σχήμα μετάδοσης μηνύματος  $\Pi'$  πετυχαίνει *authenticated communication* αν για όλους τους PPT αντιπάλους  $\mathcal{A}$  υπάρχει μια αμελητέα συνάρτηση *negl* τέτοια ώστε:

$$\Pr[\text{Auth}_{\mathcal{A},\Pi'}(n) = 1] \leq \text{negl}(n)$$

# Ανάλυση ασφάλειας

## Ορισμός

Ένα σχήμα μετάδοσης μηνύματος ( $Gen'$ ,  $EncMac'$ ,  $Dec'$ ) είναι *ασφαλές* αν είναι CCA-secure σχήμα κρυπτογράφησης και πετυχαίνει authenticated communication

# Ανάλυση ασφάλειας Κρυπτογράφηση-και-Γνησιότητα

Για μήνυμα  $m$ , στέλνονται τα  $\langle c, t \rangle$ , όπου:

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(m)$$

# Ανάλυση ασφάλειας Κρυπτογράφηση-και-Γνησιότητα

Για μήνυμα  $m$ , στέλνονται τα  $\langle c, t \rangle$ , όπου:

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(m)$$

Δεν είναι ασφαλές, γιατί παραβιάζει την ιδιωτικότητα

# Ανάλυση ασφάλειας Κρυπτογράφηση-και-Γνησιότητα

Για μήνυμα  $m$ , στέλνονται τα  $\langle c, t \rangle$ , όπου:

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(m)$$

Δεν είναι ασφαλές, γιατί παραβιάζει την ιδιωτικότητα

**Παράδειγμα:** μπορεί το  $(Gen, Mac, Vrfy)$  να είναι ασφαλές, όπως και το  $Mac'_k = (m, Mac_k(m))$ , που δίνει όλο το  $m$ !



# Ανάλυση ασφάλειας Γνησιότητα-μετά-Κρυπτογράφηση

$$t \leftarrow \text{Mac}_{k_2}(m) \text{ και } c \leftarrow \text{Enc}_{k_1}(m||t)$$

# Ανάλυση ασφάλειας Γνησιότητα-μετά-Κρυπτογράφηση

$$t \leftarrow \text{Mac}_{k_2}(m) \text{ και } c \leftarrow \text{Enc}_{k_1}(m||t)$$

Δεν είναι απαραίτητα ασφαλές

# Ανάλυση ασφάλειας Γνησιότητα-μετά-Κρυπτογράφηση

Μη ασφαλές, αν ο αντίπαλος μπορεί να μάθει αν ένα κρυπτοκείμενο είναι έγκυρο ή μη (padding attack)

# Ανάλυση ασφάλειας Γνησιότητα-μετά-Κρυπτογράφηση

Μη ασφαλές, αν ο αντίπαλος μπορεί να μάθει αν ένα κρυπτοκείμενο είναι έγκυρο ή μη (padding attack)

Στέλνει κρυπτοκείμενα και παρατηρεί την αντίδραση των τίμιων παικτών

# Ανάλυση ασφάλειας Γνησιότητα-μετά-Κρυπτογράφηση

Μη ασφαλές, αν ο αντίπαλος μπορεί να μάθει αν ένα κρυπτοκείμενο είναι έγκυρο ή μη (padding attack)

Στέλνει κρυπτοκείμενα και παρατηρεί την αντίδραση των τίμιων παικτών

Στο SSL είναι ασφαλές

# Ανάλυση ασφάλειας Κρυπτογράφηση-μετά-Γνησιότητα

Μεταδίδεται το  $\langle c, t \rangle$ , όπου

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(c)$$

## Θεώρημα

Εστω  $\Pi_E$  ένα ασφαλές σχήμα κρυπτογράφησης και  $\Pi_M$  ένα ασφαλές σχήμα γνησιότητας μηνύματος. Τότε ο συνδυασμός  $\Pi' = (Gen', EncMac', Dec')$  που παράγεται εφαρμόζοντας πρώτα κρυπτογράφηση και μετά γνησιότητα μηνύματος στα  $\Pi_E, \Pi_M$  είναι ένα ασφαλές σχήμα μετάδοσης μηνύματος.

# Ανάλυση ασφάλειας Κρυπτογράφηση-μετά-Γνησιότητα

Μεταδίδεται το  $\langle c, t \rangle$ , όπου

$$c \leftarrow Enc_{k_1}(m) \text{ και } t \leftarrow Mac_{k_2}(c)$$

## Θεώρημα

Εστω  $\Pi_E$  ένα ασφαλές σχήμα κρυπτογράφησης και  $\Pi_M$  ένα ασφαλές σχήμα γνησιότητας μηνύματος. Τότε ο συνδυασμός  $\Pi' = (Gen', EncMac', Dec')$  που παράγεται εφαρμόζοντας πρώτα κρυπτογράφηση και μετά γνησιότητα μηνύματος στα  $\Pi_E, \Pi_M$  είναι ένα ασφαλές σχήμα μετάδοσης μηνύματος.

π.χ. CCM (CTR+CBC-MAC)

# Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

Ίδιο κλειδί μπορεί να οδηγήσει σε μη ασφαλή συνδυασμό



## Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

Ίδιο κλειδί μπορεί να οδηγήσει σε μη ασφαλή συνδυασμό

**Παράδειγμα:** έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση. Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση

## Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

Ίδιο κλειδί μπορεί να οδηγήσει σε μη ασφαλή συνδυασμό

**Παράδειγμα:** έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση. Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση

Για  $Enc_k(m) = F_k(m||r)$ , όπου  $m \in \{0, 1\}^{n/2}$ ,  $r \leftarrow \{0, 1\}^{n/2}$  και  $Mac_k(c) = F_k^{-1}(c)$ , στη προσέγγιση Κρυπτογράφηση-μετά-Γνησιότητα, θα είχαμε:

## Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

Ίδιο κλειδί μπορεί να οδηγήσει σε μη ασφαλή συνδυασμό

**Παράδειγμα:** έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση. Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση

Για  $Enc_k(m) = F_k(m||r)$ , όπου  $m \in \{0, 1\}^{n/2}$ ,  $r \leftarrow \{0, 1\}^{n/2}$  και  $Mac_k(c) = F_k^{-1}(c)$ , στη προσέγγιση Κρυπτογράφηση-μετά-Γνησιότητα, θα είχαμε:

$$Enc_k(m), Mac_k(Enc_k(m)) \Rightarrow F_k(m||r), F_k^{-1}(F_k(m||r)) \Rightarrow F_k(m||r), m||r$$

## Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

Ίδιο κλειδί μπορεί να οδηγήσει σε μη ασφαλή συνδυασμό

**Παράδειγμα:** έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση. Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση

Για  $Enc_k(m) = F_k(m||r)$ , όπου  $m \in \{0, 1\}^{n/2}$ ,  $r \leftarrow \{0, 1\}^{n/2}$  και  $Mac_k(c) = F_k^{-1}(c)$ , στη προσέγγιση Κρυπτογράφηση-μετά-Γνησιότητα, θα είχαμε:

$$Enc_k(m), Mac_k(Enc_k(m)) \Rightarrow F_k(m||r), F_k^{-1}(F_k(m||r)) \Rightarrow F_k(m||r), m||r$$

άρα θα αποκαλυπτόταν το  $m$ !