

# 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 CBC-MAC
- 4 HMAC
- 5 Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)
- 6 Βιβλιογραφία

- ▶ Η κρυπτογράφηση κρύβει το μήνυμα από αντίπαλο

- ▶ Η κρυπτογράφηση κρύβει το μήνυμα από αντίπαλο
- ▶ Φτάνει για να έχουμε ασφαλή επικοινωνία;

- ▶ Η κρυπτογράφηση κρύβει το μήνυμα από αντίπαλο
- ▶ Φτάνει για να έχουμε ασφαλή επικοινωνία;
- ▶ *Γνησιότητα / ακεραιότητα μηνύματος (message integrity / authentication)*

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

- ▶ Η κρυπτογράφηση κρύβει το μήνυμα από αντίπαλο
- ▶ Φτάνει για να έχουμε ασφαλή επικοινωνία;
- ▶ *Γνησιότητα / ακεραιότητα μηνύματος (message integrity / authentication)*
- ▶ Παράδειγμα: Η εταιρεία Β παίρνει παραγγελία από την εταιρεία Α να φτιάξει 1000 οθόνες  
Ερωτήματα:
  1. Το έστειλε η Α πραγματικά;

- ▶ Η κρυπτογράφηση κρύβει το μήνυμα από αντίπαλο
- ▶ Φτάνει για να έχουμε ασφαλή επικοινωνία;
- ▶ Γνησιότητα / ακεραιότητα μηνύματος (*message integrity / authentication*)
- ▶ Παράδειγμα: Η εταιρεία Β παίρνει παραγγελία από την εταιρεία Α να φτιάξει 1000 οθόνες  
Ερωτήματα:
  1. Το έστειλε η Α πραγματικά;
  2. Είναι το 1000 σωστό;



- ▶ Η κρυπτογράφηση κρύβει το μήνυμα από αντίπαλο
- ▶ Φτάνει για να έχουμε ασφαλή επικοινωνία;
- ▶ *Γνησιότητα / ακεραιότητα μηνύματος (message integrity / authentication)*
- ▶ Παράδειγμα: Η εταιρεία Β παίρνει παραγγελία από την εταιρεία Α να φτιάξει 1000 οθόνες  
Ερωτήματα:
  1. Το έστειλε η Α πραγματικά;
  2. Είναι το 1000 σωστό;
- ▶ Κρυπτογραφικές τεχνικές και εργαλεία για να εντοπίζουμε πειραγμένα μηνύματα

- ▶ Η κρυπτογράφηση κρύβει το μήνυμα από αντίπαλο
- ▶ Φτάνει για να έχουμε 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 η A πραγματικά;
  2. Είναι το 1000 σωστό;
- ▶ Κρυπτογραφικές τεχνικές και εργαλεία για να εντοπίζουμε πειραγμένα μηνύματα
- ▶ Κρυπτογράφηση  $\neq$  Ακεραιότητα
- ▶ Οι μέθοδοι κρυπτογράφησης ιδιωτικού κλειδιού που έχουμε δει δεν εξασφαλίζουν αυθεντικότητα μηνύματος

# Stream ciphers ως κώδικες γεννησιότητας

- ▶ Έστω το απλό σχήμα κρυπτογράφησης:  $c = G(k) \oplus m$ , όπου  $G$  ένας ψευδοτυχαίος γεννήτορας.

# Stream ciphers ως κώδικες γεννησιότητας

- ▶ Έστω το απλό σχήμα κρυπτογράφησης:  $c = G(k) \oplus m$ , όπου  $G$  ένας ψευδοτυχαίος γεννήτορας.
- ▶ Αυτό το σχήμα είναι εύκολο να παραχαραχθεί: αν αντιστρέψεις ένα bit του κρυπτοκειμένου, αντιστρέφεις το αντίστοιχο bit του αποκρυπτογραφημένου κειμένου.

# Stream ciphers ως κώδικες γεννησιότητας

- ▶ Έστω το απλό σχήμα κρυπτογράφησης:  $c = G(k) \oplus m$ , όπου  $G$  ένας ψευδοτυχαίος γεννήτορας.
- ▶ Αυτό το σχήμα είναι εύκολο να παραχαραχθεί: αν αντιστρέψεις ένα bit του κρυπτοκειμένου, αντιστρέφεις το αντίστοιχο bit του αποκρυπτογραφημένου κειμένου.
- ▶ Μπορεί να έχει πολύ σοβαρές συνέπειες (π.χ. μεταφορά χρημάτων από λογαριασμό σε λογαριασμό)

# Stream ciphers ως κώδικες γεννησιότητας

- ▶ Έστω το απλό σχήμα κρυπτογράφησης:  $c = G(k) \oplus m$ , όπου  $G$  ένας ψευδοτυχαίος γεννήτορας.
- ▶ Αυτό το σχήμα είναι εύκολο να παραχαραχθεί: αν αντιστρέψεις ένα bit του κρυπτοκειμένου, αντιστρέφεις το αντίστοιχο bit του αποκρυπτογραφημένου κειμένου.
- ▶ Μπορεί να έχει πολύ σοβαρές συνέπειες (π.χ. μεταφορά χρημάτων από λογαριασμό σε λογαριασμό)
- ▶ Ασφαλές ως προς μυστικότητα, αλλά όχι ως προς ακεραιότητα.

## Block ciphers ως Κώδικες Γνησιότητας

Ίδια επίθεση ισχύει και στα OFB- και CTR-mode σχήματα κρυπτογράφησης (XOR του μηνύματος με ένα ψευδοτυχαίο stream)

## Block ciphers ως Κώδικες Γνησιότητας

Ίδια επίθεση ισχύει και στα OFB- και CTR-mode σχήματα κρυπτογράφησης (XOR του μηνύματος με ένα ψευδοτυχαίο stream)

Ίσως οι ECB- ή CBC-mode κρυπτογραφήσεις να είναι πιο δύσκολο να δεχτούν επίθεση, μιας και η αποκρυπτογράφησή τους γίνεται με αντιστροφή μιας ψευδοτυχαίας μετάθεσης  $F$ , και τα  $F_k^{-1}(x)$ ,  $F_k^{-1}(x')$  δεν σχετίζονται ακόμα και αν τα  $x, x'$  διαφέρουν μόλις κατά ένα bit.



## Block ciphers ως Κώδικες Γνησιότητας

Ίδια επίθεση ισχύει και στα OFB- και CTR-mode σχήματα κρυπτογράφησης (XOR του μηνύματος με ένα ψευδοτυχαίο stream)

Ίσως οι ECB- ή CBC-mode κρυπτογραφήσεις να είναι πιο δύσκολο να δεχτούν επίθεση, μιας και η αποκρυπτογράφησή τους γίνεται με αντιστροφή μιας ψευδοτυχαίας μετάθεσης  $F$ , και τα  $F_k^{-1}(x)$ ,  $F_k^{-1}(x')$  δεν σχετίζονται ακόμα και αν τα  $x, x'$  διαφέρουν μόλις κατά ένα bit.

Ωστόσο και σε αυτές, μικρές αλλαγές στο κρυπτοκείμενο προκαλούν προβλέψιμες αλλαγές στο αρχικό μήνυμα. Ας τις δούμε:

## Block ciphers ως Κώδικες Γνησιότητας

Ίδια επίθεση ισχύει και στα OFB- και CTR-mode σχήματα κρυπτογράφησης (XOR του μηνύματος με ένα ψευδοτυχαίο stream)

Ίσως οι ECB- ή CBC-mode κρυπτογραφήσεις να είναι πιο δύσκολο να δεχτούν επίθεση, μιας και η αποκρυπτογράφησή τους γίνεται με αντιστροφή μιας ψευδοτυχαίας μετάθεσης  $F$ , και τα  $F_k^{-1}(x)$ ,  $F_k^{-1}(x')$  δεν σχετίζονται ακόμα και αν τα  $x, x'$  διαφέρουν μόλις κατά ένα bit.

Ωστόσο και σε αυτές, μικρές αλλαγές στο κρυπτοκείμενο προκαλούν προβλέψιμες αλλαγές στο αρχικό μήνυμα. Ας τις δούμε:

ECB: αλλάζοντας ένα μόνο bit στο  $i$  block της αποκρυπτογράφησης, αλλάζει μόνο το  $i$  block του μηνύματος (πιθανή επίθεση).

## Block ciphers ως Κώδικες Γνησιότητας

Ίδια επίθεση ισχύει και στα OFB- και CTR-mode σχήματα κρυπτογράφησης (XOR του μηνύματος με ένα ψευδοτυχαίο stream)

Ίσως οι ECB- ή CBC-mode κρυπτογραφήσεις να είναι πιο δύσκολο να δεχτούν επίθεση, μιας και η αποκρυπτογράφησή τους γίνεται με αντιστροφή μιας ψευδοτυχαίας μετάθεσης  $F$ , και τα  $F_k^{-1}(x)$ ,  $F_k^{-1}(x')$  δεν σχετίζονται ακόμα και αν τα  $x, x'$  διαφέρουν μόλις κατά ένα bit.

Ωστόσο και σε αυτές, μικρές αλλαγές στο κρυπτοκείμενο προκαλούν προβλέψιμες αλλαγές στο αρχικό μήνυμα. Ας τις δούμε:

ECB: αλλάζοντας ένα μόνο bit στο  $i$  block της αποκρυπτογράφησης, αλλάζει μόνο το  $i$  block του μηνύματος (πιθανή επίθεση).

Μπορείς να αλλάξεις τη σειρά των block του ciphertext αλλάζοντας έτσι τη σειρά των block του plaintext ή να πετάξεις τα τελευταία block του ciphertext άρα και του plaintext.

## Block ciphers ως Κώδικες Γνησιότητας

Ίδια επίθεση ισχύει και στα OFB- και CTR-mode σχήματα κρυπτογράφησης (XOR του μηνύματος με ένα ψευδοτυχαίο stream)

Ίσως οι ECB- ή CBC-mode κρυπτογραφήσεις να είναι πιο δύσκολο να δεχτούν επίθεση, μιας και η αποκρυπτογράφησή τους γίνεται με αντιστροφή μιας ψευδοτυχαίας μετάθεσης  $F$ , και τα  $F_k^{-1}(x)$ ,  $F_k^{-1}(x')$  δεν σχετίζονται ακόμα και αν τα  $x, x'$  διαφέρουν μόλις κατά ένα bit.

Ωστόσο και σε αυτές, μικρές αλλαγές στο κρυπτοκείμενο προκαλούν προβλέψιμες αλλαγές στο αρχικό μήνυμα. Ας τις δούμε:

ECB: αλλάζοντας ένα μόνο bit στο  $i$  block της αποκρυπτογράφησης, αλλάζει μόνο το  $i$  block του μηνύματος (πιθανή επίθεση).

Μπορείς να αλλάξεις τη σειρά των block του ciphertext αλλάζοντας έτσι τη σειρά των block του plaintext ή να πετάξεις τα τελευταία block του ciphertext άρα και του plaintext.

CBC: αλλάζοντας το  $j$ -οστό bit του IV, αλλάζει μόνο το  $j$ -οστό bit του πρώτου block του μηνύματος (επικίνδυνο, συνήθως header)

# Message Authentication Code (MAC)

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

# Message Authentication Code (MAC)

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

# Message Authentication Code (MAC)

- ▶ Δύο παίκτες θέλουν να επικοινωνήσουν ανταλλάσσοντας ‘ακέραια / γνήσια μηνύματα’
- ▶ Ανταλλάσσουν ένα 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 / γνήσια μηνύματα’
- ▶ Ανταλλάσσουν ένα 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 / γνήσια μηνύματα’
- ▶ Ανταλλάσσουν ένα 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 ένα bit  $b = 1$ , αν η ετικέτα είναι έγκυρη, αλλιώς  $b = 0$  (ντετερμινιστικός αλγόριθμος). Δηλ.  $b = Vrfy_k(m, t)$ .

Ορθότητα: 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 προσαρμόζει τα μηνύματα που στέλνει

## 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  $\text{Mac}_k()$ . 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:

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

## 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 (εξαρτάται 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 το  $t$  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 το  $t$  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 το  $t$  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 κώδικας γνησιότητας για μηνύματα καθορισμένου 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  $n$
- ▶ 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  $n$
- ▶ 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  $n$
- ▶ 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. 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  $n$
- ▶ 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. 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  $n$
- ▶ 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. Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
    - 2.1 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  $n$
- ▶ 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. Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
    - 2.1 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  $n$
- ▶ 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. Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
    - 2.1 Πάρε κάθε block ξεχωριστά,  $t := \langle MAC'_k(m_1), \dots, MAC'_k(m_d) \rangle$   
Αν αλλάξεις σειρά στα blocks;
    - 2.2 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  $n$
- ▶ 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. Σπάμε το μήνυμα  $m$  σε  $m_1, \dots, m_d$  blocks και βρίσκουμε τη γνησιότητα κάθε block με κάποιο τρόπο
    - 2.1 Πάρε κάθε block ξεχωριστά,  $t := \langle MAC'_k(m_1), \dots, MAC'_k(m_d) \rangle$   
Αν αλλάξεις σειρά στα blocks;
    - 2.2 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)  $Vrfy'_k(r || l || i || m_i) = t_i$ , για όλα τα  $1 \leq i \leq d$

# 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*

# CBC-MAC

Μπορούμε να φτιάξουμε MAC με χρήση block cipher.

Βασική έκδοση (μη αποδοτική):

## Κατασκευή CBC-MAC

Έστω  $F$  μια ψευδοτυχαία συνάρτηση και  $l > 0$  μια συνάρτηση μήκους. Η βασική CBC-MAC είναι:

- ▶ *Mac*: με είσοδο  $k \in \{0, 1\}^n$  και  $m$  μήκους  $l(n)n$  κάνε τα εξής:
  1. Γράψε το  $m$  ως  $m = m_1, \dots, m_l$ , όπου κάθε  $m_i$  έχει μήκος  $n$ .
  2. Θέσε  $t_0 = 0^n$ . Τότε, για  $i = 1, \dots, l$ : κάνε  $t_i = F_k(t_{i-1} \oplus m_i)$, δώσε έξοδο το  $t_l$ .
- ▶ *Verfy*: με είσοδο  $k, m, t$ , αν το  $m$  δεν είναι μήκους  $l(n)n$ , δώσε αποτέλεσμα 0, αλλιώς 1, αν  $t \stackrel{?}{=} Mac_k(m)$ .

## Θεώρημα

*Εστω  $l$  πολυώνυμο. Αν η  $F$  είναι μια ψευδοτυχαία συνάρτηση, τότε η παραπάνω κατασκευή είναι ένα ασφαλές MAC για μηνύματα μήκους  $l(n)n$ .*

## Θεώρημα

*Εστω  $l$  πολυώνυμο. Αν η  $F$  είναι μια ψευδοτυχαία συνάρτηση, τότε η παραπάνω κατασκευή είναι ένα ασφαλές MAC για μηνύματα μήκους  $l(n)n$ .*

Παρατήρηση: Ασφαλές μόνο όταν το μήκος είναι σταθερό και προκαθορισμένο από αποστελέα και παραλήπτη.



# CBC-MAC

CBC-MAC vs CBC-mode:

1. Το CBC-mode χρησιμοποιεί τυχαίο IV, ενώ το CBC-MAC όχι (ή  $IV=0^n$ ). Με τυχαίο IV δεν είναι ασφαλές.
2. Στο CBC-mode όλα τα ενδιάμεσα  $c_i$  ( $t_i$ ) αποκαλύπτονται, στο CBC-MAC μόνο το τελευταίο. Αν αποκαλυπτόταν τα ενδιάμεσα, δεν θα ήταν ασφαλές.

## CBC-MAC vs CBC-mode:

1. Το CBC-mode χρησιμοποιεί τυχαίο IV, ενώ το CBC-MAC όχι (ή  $IV=0^n$ ). Με τυχαίο IV δεν είναι ασφαλές.
2. Στο CBC-mode όλα τα ενδιάμεσα  $c_i$  ( $t_i$ ) αποκαλύπτονται, στο CBC-MAC μόνο το τελευταίο. Αν αποκαλυπτόταν τα ενδιάμεσα, δεν θα ήταν ασφαλές.

Επέκταση της προηγούμενης κατασκευής, ώστε να δουλεύει για οποιοδήποτε μήκος μηνύματος:

1. Βάλε στην αρχή το μήκος του μηνύματος.
2. Χρησιμοποίησε δύο ανεξάρτητα, ομοιομόρφα τυχαία κλειδιά  $k_1, k_2$  και υπολόγισε το CBC-MAC του μηνύματος  $t = Mac_{k_1}(m)$  και μετά  $t' = F_{k_2}(t)$ .

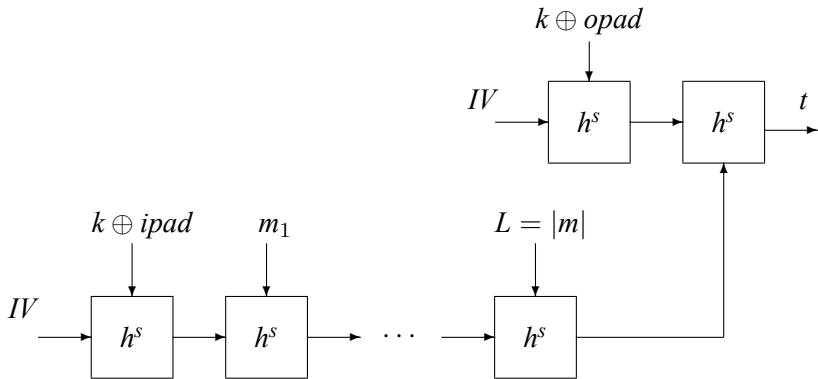
# Hashed MAC (HMAC)

Μπορούμε να χρησιμοποιήσουμε συνάρτηση σύνοψης για MAC:

## 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  $t \stackrel{?}{=} H^s((k \oplus opad) || H^s(k \oplus ipad) || m)$



Σχήμα : HMAC

## Σημασία των ipad και opad

- ▶ Η ασφάλεια του HMAC μπορεί να βασιστεί στην ασθενέστερη υπόθεση της ασφάλειας αντίστασης δεύτερου ορίσματος (second pre-image resistance)

## Σημασία των ipad και opad

- ▶ Η ασφάλεια του HMAC μπορεί να βασιστεί στην ασθενέστερη υπόθεση της ασφάλειας αντίστασης δεύτερου ορίσματος (second pre-image resistance)
- ▶ Αρκεί ένα κλειδί (γιατί;)

# HMAC στην πράξη

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

# HMAC στην πράξη

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



# HMAC στην πράξη

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

# HMAC στην πράξη

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

# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

Έστω  $\Pi = (Gen, Enc, Dec)$  σχήμα κρυπτογράφησης, PPT αντίπαλος  $\mathcal{A}$ ,  $n$  παράμετρος ασφάλειας.

# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

Έστω  $\Pi = (Gen, Enc, Dec)$  σχήμα κρυπτογράφησης, PPT αντίπαλος  $\mathcal{A}$ ,  $n$  παράμετρος ασφάλειας.

## Πείραμα ασφαλούς μετάδοσης / μη παραχάραξης μηνύματος

**Enc-Forge** $_{\mathcal{A}, \Pi}(n)$ :

1. Ο  $Gen(1^n)$  παράγει ένα κλειδί  $k$ .
2. Ο αντίπαλος  $\mathcal{A}$  παίρνει σαν είσοδο το  $1^n$  και πρόσβαση σε ένα μαντείο  $Enc_k$ . Δίνει έξοδο ένα  $c$ .
3. Έστω  $m := Dec_k(c)$  και  $Q$  το σύνολο των ερωτήσεων που κάνει στο μαντείο. Η έξοδος είναι 1, ανν (1)  $m \neq \perp$  και (2)  $m \notin Q$

# Ορισμός ασφάλειας

## Ορισμός

Ένα σχήμα κρυπτογράφησης  $\Pi$  είναι *μη-παραχαράξιμο* αν για όλους τους PPT αντιπάλους  $\mathcal{A}$  υπάρχει μια αμελητέα συνάρτηση *negl* τέτοια ώστε:

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

# Ορισμός ασφάλειας

## Ορισμός

Ένα σχήμα κρυπτογράφησης που είναι CCA-secure σχήμα κρυπτογράφησης και μη-παραχαράξιμο είναι σχήμα που πετυχαίνει *authenticated encryption*.

# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

- ▶ Στις περισσότερες online εργασίες, ιδίως τραπεζικές συναλλαγές χρειάζονται και τα δύο

# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

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



# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

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

# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

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

# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

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

# Κρυπτογράφηση και γνησιότητα (Authenticated Encryption)

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

# Ασφαλείς συνδυασμός

- ▶ Θέλουμε οποιοσδήποτε συνδυασμός ασφαλών σχημάτων κρυπτογράφησης και MACs να είναι ασφαλής

# Ασφαλείς συνδυασμός

- ▶ Θέλουμε οποιοσδήποτε συνδυασμός ασφαλών σχημάτων κρυπτογράφησης και MACs να είναι ασφαλής
- ▶ Έτσι, ώστε να ελαχιστοποιήσουμε λάθη στην υλοποίηση (αντικατάσταση με νεότερες εκδόσεις ή αλλαγή των standards)

# Ασφαλείς συνδυασμός

- ▶ Θέλουμε οποιοσδήποτε συνδυασμός ασφαλών σχημάτων κρυπτογράφησης και MACs να είναι ασφαλής
- ▶ Έτσι, ώστε να ελαχιστοποιήσουμε λάθη στην υλοποίηση (αντικατάσταση με νεότερες εκδόσεις ή αλλαγή των standards)
- ▶ Σημείωση: Μπορεί να υπάρχει ένας συνδυασμός σχημάτων που είναι ασφαλής (π.χ. SSH, IPsec, SSL)

## Συνδυασμός σχημάτων

- ▶ Έστω  $\Pi_E = (Enc, Dec)$ , ένα CPA-secure σχήμα κρυπτογράφησης και  $\Pi_M = (Mac, Vrfy)$ , ένα σχήμα MAC
- ▶ Δύο ανεξάρτητα κλειδιά  $k_E, k_M$  αντίστοιχα, που παράγονται από την  $Gen$ , για κάθε παράμετρο ασφάλειας  $n$



## Συνδυασμός σχημάτων

- ▶ Έστω  $\Pi_E = (Enc, Dec)$ , ένα CPA-secure σχήμα κρυπτογράφησης και  $\Pi_M = (Mac, Vrfy)$ , ένα σχήμα MAC
- ▶ Δύο ανεξάρτητα κλειδιά  $k_E, k_M$  αντίστοιχα, που παράγονται από την  $Gen$ , για κάθε παράμετρο ασφάλειας  $n$

## Συνδυασμός σχημάτων

- ▶ Έστω  $\Pi_E = (Enc, Dec)$ , ένα CPA-secure σχήμα κρυπτογράφησης και  $\Pi_M = (Mac, Vrfy)$ , ένα σχήμα MAC
- ▶ Δύο ανεξάρτητα κλειδιά  $k_E, k_M$  αντίστοιχα, που παράγονται από την  $Gen$ , για κάθε παράμετρο ασφάλειας  $n$

### 1. Κρυπτογράφηση-και-γνησιότητα

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(m)$$

## Συνδυασμός σχημάτων

- ▶ Έστω  $\Pi_E = (Enc, Dec)$ , ένα CPA-secure σχήμα κρυπτογράφησης και  $\Pi_M = (Mac, Vrfy)$ , ένα σχήμα MAC
- ▶ Δύο ανεξάρτητα κλειδιά  $k_E, k_M$  αντίστοιχα, που παράγονται από την  $Gen$ , για κάθε παράμετρο ασφάλειας  $n$

### 1. Κρυπτογράφηση-και-γνησιότητα

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(m)$$

### 2. Γνησιότητα-μετά-κρυπτογράφηση

$$t \leftarrow Mac_{k_M}(m) \text{ και } c \leftarrow Enc_{k_E}(m||t)$$

## Συνδυασμός σχημάτων

- ▶ Έστω  $\Pi_E = (Enc, Dec)$ , ένα CPA-secure σχήμα κρυπτογράφησης και  $\Pi_M = (Mac, Vrfy)$ , ένα σχήμα MAC
- ▶ Δύο ανεξάρτητα κλειδιά  $k_E, k_M$  αντίστοιχα, που παράγονται από την  $Gen$ , για κάθε παράμετρο ασφάλειας  $n$

### 1. Κρυπτογράφηση-και-γνησιότητα

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(m)$$

### 2. Γνησιότητα-μετά-κρυπτογράφηση

$$t \leftarrow Mac_{k_M}(m) \text{ και } c \leftarrow Enc_{k_E}(m||t)$$

### 3. Κρυπτογράφηση-μετά-γνησιότητα

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(c)$$

# Ανάλυση ασφάλειας Κρυπτογράφηση-και-Γνησιότητα

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

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(m)$$

# Ανάλυση ασφάλειας Κρυπτογράφηση-και-Γνησιότητα

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

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(m)$$

Δεν είναι ασφαλές, γιατί παραβιάζει την ιδιωτικότητα

# Ανάλυση ασφάλειας Κρυπτογράφηση-και-Γνησιότητα

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

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(m)$$

Δεν είναι ασφαλές, γιατί παραβιάζει την ιδιωτικότητα

**Παράδειγμα:** μπορεί το MAC να είναι ασφαλές, αλλά να δίνει π.χ. το 1ο bit του  $t$  να είναι ίδιο με του μηνύματος (παραβιάζει indistinguishability)

# Ανάλυση ασφάλειας Γνησιότητα-μετά-Κρυπτογράφηση

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

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

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



# Ανάλυση ασφάλειας Γνησιότητα-μετά-Κρυπτογράφηση

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

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

- ▶ 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

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

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

- ▶ 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

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

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(c)$$

# Ανάλυση ασφάλειας Κρυπτογράφηση-μετά-Γνησιότητα

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

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(c)$$

Είναι ασφαλές, όπως αποδεικνύει το θεώρημα:

## Θεώρημα

*Έστω  $\Pi_E$  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

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

$$c \leftarrow Enc_{k_E}(m) \text{ και } t \leftarrow Mac_{k_M}(c)$$

Είναι ασφαλές, όπως αποδεικνύει το θεώρημα:

## Θεώρημα

*Έστω  $\Pi_E$  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

## Παράδειγμα

Έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση.

Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση.

# Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

## Παράδειγμα

Έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση.

Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση.

Έστω  $Enc_k(m) = F_k(m||r)$ , όπου  $m \in \{0, 1\}^{n/2}$ ,  $r \leftarrow \{0, 1\}^{n/2}$  και

$Mac_k(c) = F_k^{-1}(c)$ .

Στην προσέγγιση Κρυπτογράφηση-μετά-Γνησιότητα, θα είχαμε:

# Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

## Παράδειγμα

Έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση.

Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση.

Έστω  $Enc_k(m) = F_k(m||r)$ , όπου  $m \in \{0, 1\}^{n/2}$ ,  $r \leftarrow \{0, 1\}^{n/2}$  και

$Mac_k(c) = F_k^{-1}(c)$ .

Στην προσέγγιση Κρυπτογράφηση-μετά-Γνησιότητα, θα είχαμε:

$$\langle c, t \rangle = \langle F_k(m||r), F_k^{-1}(F_k(m||r)) \rangle = \langle F_k(m||r), m||r \rangle$$



# Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

## Παράδειγμα

Έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση.

Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση.

Έστω  $Enc_k(m) = F_k(m||r)$ , όπου  $m \in \{0, 1\}^{n/2}$ ,  $r \leftarrow \{0, 1\}^{n/2}$  και

$Mac_k(c) = F_k^{-1}(c)$ .

Στην προσέγγιση Κρυπτογράφηση-μετά-Γνησιότητα, θα είχαμε:

$$\langle c, t \rangle = \langle F_k(m||r), F_k^{-1}(F_k(m||r)) \rangle = \langle F_k(m||r), m||r \rangle$$

άρα θα αποκαλυπτόταν το  $m$ !

# Διαφορετικά κλειδιά για διαφορετικές εφαρμογές

## Παράδειγμα

Έστω  $F$  μια ισχυρή ψευδοτυχαία μετάθεση.

Τότε και η  $F^{-1}$  είναι ισχυρή ψευδοτυχαία μετάθεση.

Έστω  $Enc_k(m) = F_k(m||r)$ , όπου  $m \in \{0, 1\}^{n/2}$ ,  $r \leftarrow \{0, 1\}^{n/2}$  και

$Mac_k(c) = F_k^{-1}(c)$ .

Στην προσέγγιση Κρυπτογράφηση-μετά-Γνησιότητα, θα είχαμε:

$$\langle c, t \rangle = \langle F_k(m||r), F_k^{-1}(F_k(m||r)) \rangle = \langle F_k(m||r), m||r \rangle$$

άρα θα αποκαλυπτόταν το  $m$ !

Συμπέρασμα: Ίδιο κλειδί μπορεί να οδηγήσει σε μη ασφαλή συνδυασμό.

# Βιβλιογραφία

- ▶ Jonathan Katz and Yehuda Lindell. Introduction to Modern Cryptography (Chapman and Hall/Crc Cryptography and Network Security Series). Chapman and Hall/CRC, 2007
- ▶ M. Bellare, J. Kilian, and P. Rogaway. The security of the cipher block chaining message authentication code. Journal of Computer and System Sciences, 61(3):362-399, 2000.
- ▶ S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. SIAM J. Computing, 17(2):281-308, 1988.
- ▶ M. Bellare, R. Canetti, and H. Krawczyk. Keying hash functions for message authentication. Advances in Cryptology - Crypto 96 Proceedings, Lecture Notes in Computer Science Vol. 1109, N. Koblitz ed, Springer-Verlag, 1996.