UNIVERSITÀ DEGLI STUDI "ROMA TRE"
CORSO DI STUDI IN SCIENZE COMPUTAZIONALI
IN450 - ALGORITMI PER LA CRITTOGRAFIA – A.A. 2020-2021

M. PEDICINI

ESONERO 2 – 08/06/2021 – TEMPO 3H00

COGNOME _____ NOME _____ MATRICOLA _____

Esercizio 1. *Considerare un crittosistema con insieme dei plaintext $P = \{a, b, c\}$, insieme delle chiavi $K = \{k_1, k_2, k_3, k_4\}$ e insieme dei ciphertext $C = \{1, 2, 3, 4, 5\}$ con la seguente matrice di cifratura:*

$$M = \begin{array}{|c|ccc|} \hline & a & b & c \\ \hline k_1 & 1 & 3 & 4 \\ k_2 & 4 & 1 & 5 \\ k_3 & 1 & 5 & 3 \\ k_4 & 2 & 5 & 3 \\ \hline \end{array}$$

e le seguenti distribuzioni di probabilità: $p(k_1) = p(k_2) = 3/10$, $p(k_3) = 1/15$, $p(k_4) = 1/3$ e $p(a) = 1/4$, $p(b) = 1/8$, $p(c) = 5/8$.

(1) Calcolare la distribuzione di probabilità indotta su C;
(2) calcolare l'entropia $H(P)$;
(3) mediante l'algoritmo di Huffman ricavare un codice binario per rappresentare in modo efficiente gli elementi di C;
(4) calcolare l'entropia condizionata $H(K|C)$.

Esercizio 2. *Una permutazione σ sull'insieme della parole binarie a n-bit $S = \{0, 1\}^n$, è un* ortomorfismo *quando σ' definita come*

$$\sigma'(a) = a \oplus \sigma(a) \qquad \text{è ancora una permutazione}$$

dove $\oplus$ rappresenta lo XOR *bit-a-bit.*

Fissato $n = 8$, consideriamo la funzione

$$\omega(a) := \mathsf{ROT}^4(a \oplus (a >> 4))$$

dove $>>$ i denota lo shift di i bit a destra (senza reinserimento) e ROT^i denota la rotazione di i bit a destra.

(1) dimostrare che ω è una permutazione (iniettività sulle 2^n parole binarie);
(2) dimostrare che ω è un ortomorfismo;
(3) disegnare un diagramma che descrive ω;
(4) calcolare l'inversa di ω;
(5) si consideri ora la funzione

$$\pi(a) := (a \,\mathsf{AND}\, c) \oplus \mathsf{ROT}^1(a)$$

dove $c =$ `0x55`*, in modo analogo ai punti (1) (2) (3) e (4) mostrare che π è un ortomorfismo e calcolarne l'inversa.*

(6) *calcolare la riga della tabella di approssimazione lineare di* π *corrispondente all'indice* `0x03`.

Esercizio 3. *(KeyExpansion/AES-SubBytes) Sia* $p(x) = x^3 + x^2 + 1$ *si consideri la S-box di AES/Rijndael adattata opportunamente a 3 bit e che incorpora le operazioni sul campo finito* $\mathbb{F}_{2^3} = \mathbb{F}_2[x]/p(x)$.

Eseguire un round di cifratura di AES del blocco $X =$ `0x04A2982522D8` *ponendo la chiave* $k =$ `0xDEC1DEADA` *(*`0x` *è la notazione per indicare il sistema di numerazione esadecimale):*

(1) *Adattare opportunamente l'algoritmo AES alle dimensioni del campo discutendo la scelta effettuata (in particolare nel caso della* SubBytes*).*

Nel seguito le funzioni SubBytes, ShiftRows, MixColumns *e* AddKey *sono quelle dell'algoritmo adattato:*

a. *Calcolare* $S_1 = \mathsf{SubBytes}(X)$*;*
b. *Calcolare* $S_2 = \mathsf{ShiftRows}(S_1)$*;*
c. *Calcolare* $S_3 = \mathsf{MixColumns}(S_2)$*;*
d. *Calcolare* $S_4 = \mathsf{AddKey}(S_3, key_0)$ *dove* key_0 *è il primo elemento del* KeySchedule *calcolato a partire dalla chiave* k.