Corso di Crittografia
Anno Accademico 2018-2019
Homework I
15 Marzo 2019

Si ricorda che:

- Gli esercizi si risolvono individualmente.
- Le soluzioni vanno scritte in LaTeXe inviate al docente, in formato `pdf`, all'indirizzo di posta elettronica `ugo.dallago@unibo.it`. Vi chiedo la cortesia di definire l'oggetto della mail come "`[Crittografia] Consegna Homework I`". Si invitano gli studenti ad utilizzare il template `http://www.cs.unibo.it/~dallago/CRI1819/Homework-template-1819.tex`.
- La scadenza per l'invio delle soluzioni è il 24 di Marzo alle ore 24.00 CET.

Esercizio 1.
Si consideri il cifrario a sostituzione polialfabetica e lo si veda come un cifrario nel senso della *seconda* definizione vista a lezione, quella più concreta basata su algoritmi PPT che lavorano su stringhe binarie.

- Si dia una definizione formale del cifrario così ottenuto.
- Si dimostri che tale cifrario *non* è sicuro contro attacchi passivi. Nel fare ciò si diano opportune condizioni sulla cardinalità dell'alfabeto Σ su cui lavora il cifrario a sostituzione polialfabetica e sulla lunghezza del periodo. Cosa si può dire quando tali condizioni non valgono? Cosa diventa in tal caso il cifrario a sostituzione polialfabetica?

Esercizio 2.
Un programmatore vorrebbe costruire uno schema di codifica a partire da un generatore pseudocasuale G a lunghezza fissa ℓ. Per qualche strano motivo, però, l'output di G è troppo lungo per le esigenze del programmatore. Si cerchi di convincere il programmatore che quello che sta facendo non pregiudica la sicurezza di G. Più concretamente, si definisca, per ogni $\xi : \mathbb{N} \to \mathbb{N}$ tale che $\ell(n) \geq n + 1 + \xi(n)$, l'algoritmo G^ξ ponendo $G^\xi(x) = b_1 \cdots b_{\ell(|x|)-\xi(|x|)}$ ogniqualvolta $G(x) = b_1 \cdots b_{\ell(|x|)}$. Si dimostri che G^ξ è pseudocasuale ogniqualvolta G è pseudocasuale. Che altre operazioni di "postprocessing" ha senso suggerire al programmatore? Permutazioni dei bit? Duplicazioni di bit?

Esercizio 3.
Ad una banca viene proposto di utilizzare uno schema di codifica sicuro rispetto ad attacchi passivi per cifrare i dati relativi alle transazioni dei suoi clienti. I funzionari di questa banca, però, non sono molto convinti del fatto che la definizione basata sull'esperimento $\mathsf{PrivK}^{\mathsf{eav}}$ garantisca la confidenzialita dei dati inviati. In particolare, non sono affatto convinti che tale definizione garantisca che nessun avversario possa dedurre a partire da un crittogramma c, il valore di uno qualunque degli ultimi 16 bit del sottostante messaggio in chiaro m (se non con probabilità molto bassa). Si introduca una nuova nozione di cifrario sicuro (chiamiamola *segretezza degli ultimi* 16 *bit*) che catturi esattamente la nozione di sicurezza che la banca vuole raggiungere. Si provi, infine, che ogni schema sicuro rispetto ad attacchi passivi garantisce la segretezza degli ultimi 16 bit.