Autor: Ruxandra F. Olimid
Departamentul de Informatică, Universitatea din București

- Laboratorul 6 -

Sisteme de criptare simetrice

Disclaimer: Pe parcursul acestui curs/laborator vi se vor prezenta diverse noțiuni de securitate informatică, cu scopul de a învăța cum să securizați sistemele. Toate noțiunile și exercițiile sunt prezentate în scop didactic, chiar dacă uneori se presupune să gândiți ca un adversar. Nu folosiți aceste tehnici în scopuri malițioase! Acestea pot avea consecințe legale în cazul comiterii unor infracțiuni, pentru care **deveniți pe deplin răspunzători**!

1. *Linear Feedback Shift Registers (LFSRs)*


Citiți despre *Linear Feedback Shift Register (LFSR)* în [1], pg.355-358.


Implementați un LFSR definit peste $\mathbf{F_2}$ care permite citirea de la tastatură a coeficienților c_1, ... c_L și a stării inițiale s_0, ..., s_{L-1} (toate aceste valori sunt elemente binare, 0 sau 1). Generați și afișați secvența de ieșire pentru o perioadă completă și valoarea perioadei. Testați funcționarea corectă pe exemplul din Tabelul 1 prezent în sursa indicată mai sus.

Notă: Puteți realiza implementarea în orice limbaj de programare convenabil, dar trebuie să puteți explica codul sursă profesorului de laborator.

2. *Advanced Encryption Standard (AES)*


Citiți despre utilizarea AES în Python [2].


Se dă următoarea secvență de instrucțiuni:

```
from Crypto.Cipher import AES

key = b'O cheie oarecare'
data=b'testtesttesttesttesttesttesttesttesttesttesttest'

cipher = AES.new(key, AES.MODE_ECB)
cipher.encrypt(data)
```

Răspundeți la următoarele cerințe:

a) Executați secvența de mai sus. Ce obțineți?
b) Ce mod de operare este folosit? Ce observați?
c) Ați recomanda folosirea modului de operare de la b)? De ce? De ce nu?
d) Care este dimensiunea cheii? Dar a blocului?
e) Modificați codul astfel încât să funcționeze dacă se înlocuiește valoarea `data` cu `data=b'test'`.
f) Refaceți codul, schimbând modul de operare cu un alt mod de operare pe care îl considerați mai potrivit.

3. *Atacul Meet-in-the-Middle*

Înțelegeți cum funcționează *Data Encryption Standard (DES)* și atacul *Meet-in-the-Middle* citind [1], pg.129-133 și vizionând [3,4]. Înțelegeți de ce *Triple—DES* este folosit, în timp ce „*Double-DES"* nu aduce beneficii majore.

Se dă următoarea secvență de instrucțiuni, pentru care `?` din `key1` și `key2` reprezintă o cifra hexazecimală necunoscută:

```
from Crypto.Cipher import DES

key1 = '\x?0\x00\x00\x00\x00\x00\x00\x00'
key2 = '\x?0\x00\x00\x00\x00\x00\x00\x00'

cipher1 = DES.new(key1, DES.MODE_ECB)
cipher2 = DES.new(key2, DES.MODE_ECB)

plaintext = "Provocare MitM!!"
ciphertext = cipher2.encrypt(cipher1.encrypt(plaintext))
```

În urma execuției, se obține:

```
ciphertext = "G\xfd\xdfpd\xa5\xc9'C\xe2\xf0\x84)\xef\xeb\xf9"
```

Implementați un atac de tip *Meet-in-the-Middle* pentru a determina cele 2 chei (i.e., cele doua valori hexazecimale marcate cu `?`). Câte chei ați testat în total? Câte criptări / decriptări ați făcut?

Referințe bibliografice

1. Kryszczuk, K., & Richiardi, J. (2014). *Springer Encyclopedia of Cryptography and Security*. Accesibil la: https://www.researchgate.net/publication/230674947_Springer_Encyclopedia_of_Cryptography_and_Security.
2. PyCryptodome. *AES*. https://pycryptodome.readthedocs.io/en/latest/src/cipher/aes.html
3. D.Boneh. *Cryptography – The Data Encryption Standard*. Accesibil la: https://youtu.be/l-uZvAlFEEU?list=PL2jykFOD1AWYosqucluZghEVjUkopdD1e
4. D.Boneh. *Cryptography – Exhaustive Search Attacks*. Accesibil la: https://youtu.be/V_HFxe73Kpg?list=PL2jykFOD1AWYosqucluZghEVjUkopdD1e