Securitatea Informatiei

Documentatie Tema 1

Știrbu Alexandru-Ilie [IIIB4]

November 4, 2020

1 Cum se ruleaza

Proiectul este facut în Python, în environment-ul PyCharm.

Pentru a putea rula proiectul, există 2 metode:

Pachete necesare: pycryptodome (se poate instala folosind următoarea comandă)

```
pip3 -install pycryptodome
```

Metoda 1: Încărcarea folder-ului direct în PyCharm, și apoi rularea în ordinea următoare a fișierelor: server.py pentru a descihde KeyMaster-ul, client.py pentru a deschide node-ul A, și clint.py pentru a deschide node-ul B

Metoda 2: De la linia de comandă, a se rula următoarele comenzi, în ordinea următoare:

```
python.exe ./server.py - Pentru a deschide KeyMaster-ul
python.exe ./client.py - Pentru a deschide Node-ul A
python.exe ./client.py - Pentru a deschide Node-ul B
```

2 Explicarea criptarii

Am avut de implementat 2 modalități de encriptare și decriptare: AES-CBC și AES-CFB

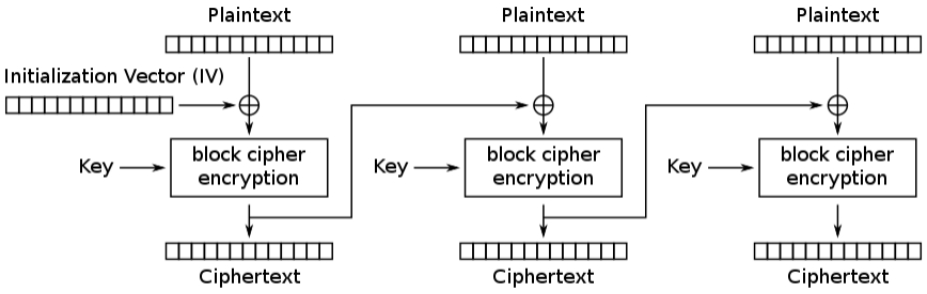
Ambele modalități, au fost implementate folosind clase, în care se salvau vectorul de inițializare (IV), cheia (KEY), și care au metode de criptare/decriptare care primesc ca parametrii un text/criptotext în format binar.

Criptarea la ambele metode împarte plain-text-ul primit în block-uri de câte 16 bytes, aplicând și o padare pentru ultimul block cu scopul de a umple block-ul cu informație. Dacă block-ul nu este umplut cu informație, padarea calculeaz nr de bytes necesar astfel încât să fie 16 bytes (X = 16-len(block)), și umple toți bytes lipsă cu chr(X). Dacă block-ul este umplut, ca să fie recunoscută padare, se crează un alt block cu 16 bytes, setați toți cu valoarea chr(16) și după se criptează text-ul paddat.

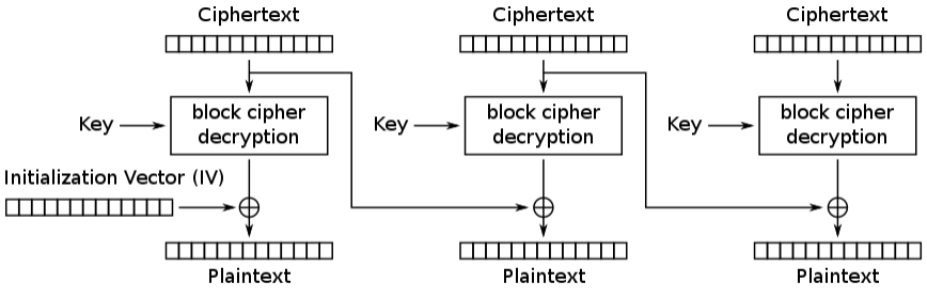
Decriptarea la ambele metode împarte cripto-text-ul în block-uri de câte 16 byte, decriptează text-ul, și aplică depadarea, care ia ultimul byte de informație (X = decrypted[-1]) pentru a elimina ultimii X bytes de infromație din textul decriptat, iar aopi se aplică decriptarea pe text-ul ne-paddat.

2.1 AES-CBC

Criptarea și decriptarea au urmat următoarele scheme de implementare:



Cipher Block Chaining (CBC) mode encryption

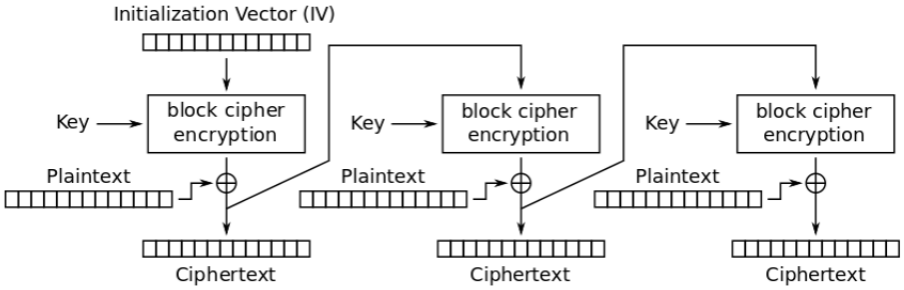


Cipher Block Chaining (CBC) mode decryption

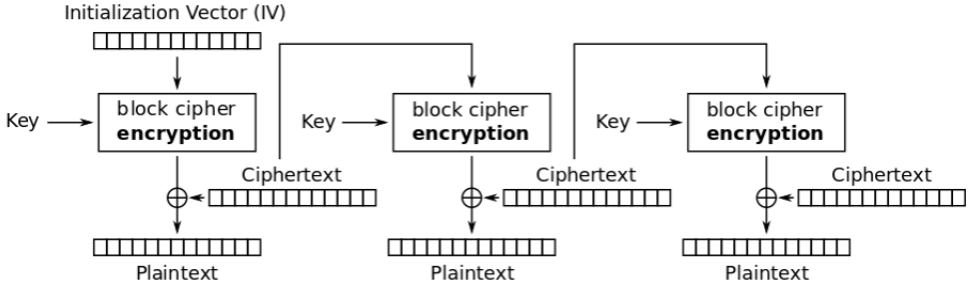
Criptarea și decriptarea pentru "block cipher encryption" și "block cipher decryption" folosesc funcțiile din pachet-ul pycryptodome implementate pentru AES-ECB.encript() și AES-ECB.decript().

2.2 AES-CFB

Criptarea și decriptarea au urmat următoarele scheme de implementare:



Cipher Feedback (CFB) mode encryption



Cipher Feedback (CFB) mode decryption

Criptarea pentru "block cipher encryption" folosește funcția din pachet-ul py-cryptodome implementat pentru AES-ECB.encript()

3 Interacțiunea dintre nodurile A B și server-ul KM

Scopul principal este ca A să-i trimită lui B un fișer ce conține un mesaj, pe care A să-l cripteze folosind o cheie și un vector de inițializare date de KM și B să poată să decripteze acel fișier criptat, folosdin aceeași cheie și vector de inițializare generate de KM.

Ordinea detaliata la evenimente este:

- KM-ul va deschide un server la port-ul 1234
- KM generează un vector de inițializare random IV, și 3 chei random K1, K2, K3 (toate de câte 16 bytes)
- Primul client care se va conecta la KM va fi considerat nodul A, și va primi un mesaj în care i se explică că identitatea lui este A
- Al 2-lea client care se va conecta la KM va fi considerat nodul B, și va primi un mesaj în care i se explică că identitatea lui este B
- Ambii clienți primesc cheia K3 necriptată
- Ambii clineți primesc vectorul de inițializare criptat prin metoda AES-ECB, folosind cheia K3
- A-ul alege la întâmplare o metodă de criptare dintre CBC / CFB
- A-ul trimite către KM metoda de criptare dorită (CBC sau CFB) criptată cu AES-ECB folosind cheia K3
- KM-ul trimite câtre B metoda de criptare doriă de A, criptată cu cheia K3
- KM-ul oferă K1 dacă metodă de criptare de CBC, sau K2 dacă metoda de criptare este CFB câtre A și B
- B-ul devine un server la portul 2345, și notifică KM că A-ul poate să se conecteze la B
- KM-ul trimite un mesaj către A, anunțându-l că se poate conecta la B
- A-ul se conectează la serverul lui B
- A-ul citește conținutul binar din fișierul "file.txt"
- A-ul criptează conținutul citit, folosind metoda de criptare aleasă, cu parametrii cheia specifică modalității de criptare (decriptată cu K3) și IV-ul (decriptat cu K3)
- A-ul trimite către B conținutul criptat

- A-ul se deconectează de la server-ul lui B
- B-ul decriptează conținutul criptat, folosind metoda dorita de A, cu cheia specifică trimisă de KM (decriptată cu K3) și IV (decriptat cu K3)
- Ambele cliente trimit feedback înapoi către KM legat de numărul de block-uri din plain text respectiv cripted text, și se deconectează de la serverul KM
- KM-ul va spune dacă numărul de block-uri coincid la criptare / decriptatre