TEMA SI 1

Cozma Ilie-Catalin

Noiembrie 08, 2020

1 Descrierea mediului de lucru utilizat

Pentru a porni aplicatia trebuie sa deschidem 4 terminale si sa rulam pe rand comenzile de mai jos . Fiecare nod A,B,KM este o instanta a clasei Node care initializeaza in constructor un socket pentru comunicarea cu server-ul si tipul nodului,dupa care se conecteazaza la adresa server-ului. Comunicarea intre cele 3 noduri se face prin intermediul server-ului server.py

```python
import socket
import sys

host = "127.0.0.1"
port = 12345

class Node:
    def __init__(self,node_type):
        self.node_type = node_type
        self.signal = True
        try:
            self.socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            self.socket.connect((host, port))
            self.socket.send(self.node_type.encode())
        except Exception as e:
            print("Could not make a connection to the server " + str(e))
            input("Press any key to quit")
            sys.exit(0)
```

1.1 Server terminal

In linia de comanda vom rula py server.py
Server-ul tine minte toate nodurile intr-un vector de noduri . Pentru ficare nod se tine minte socketul si de cate ori a trimis un mesaj la server , iar pentru fiecare numar va fi o anumita actiune . Se poate face diferenta intre noduri prin parametrul node-type care este trimis imediat dupa conectarea nodului la server.

1.2 Node A terminal

In linia de comanda vom rula py A.py

1.3 Node B terminal

In linia de comanda vom rula py B.py

1.4 Node KM terminal

In linia de comanda vom rula py KM.py

2 Descrierea modului de rezolvare a cerintei exercittiului

Dupa rularea comenzilor , nodul A va cere modul de criptare dorit pana cand se va introduce unul valid (ECB sau OFB) (Nu trebuie introdus modul de criptare valid pana cand B si KM nu au fost conectate la server !!!) , il va trimite lui B iar lui KM ii va trimite cheia in functie de modul de criptare (K1 pentru ECB , K2 pentru OFB). KM va cripta cheia ceruta cu K3 si o va trimite lui A si lui B.
$A \Longrightarrow server \Longrightarrow KM$
$A \Longrightarrow server \Longrightarrow B$
$B \Longrightarrow server \Longrightarrow KM$
$KM \Longrightarrow server \Longrightarrow A$
$KM \Longrightarrow server \Longrightarrow B$

```
C:\Users\Catal\OneDrive\Desktop\proiectSI1>py server.py
Node b'A' connected!
Node b'B' connected!
Node b'KM' connected!
[A->1]b'ECB'
[B->1]b'K1'
[KM->1]b'W\x88\\\xc7\x1c\x01\x00\xfe&q2\xa1\xebA\x0c\xb1'
```

```
C:\Users\Catal\OneDrive\Desktop\proiectSI1>py A.py
Enter encrypt mode: FFF
Invalid encryption mode!
Enter encrypt mode: ECB
[RECEIVE->1]b'W\x88\\\xc7\x1c\x01\x00\xfe&q2\xa1\xebA\x0c\xb1'
Decrypted K1 with K3 1111111111111111 ECB mode from -> b'W\x88\\\xc7\x1c\x01\x00\xfe&q2\xa1\xebA\x0c\xb1'
```

```
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Catal\OneDrive\Desktop\proiectSI1>py B.py
[RECEIVE->1]b'ECB'
[RECEIVE->2]b'W\x88\\\xc7\x1c\x01\x00\xfe&q2\xa1\xebA\x0c\xb1'
Decrypted K1 with k3 1111111111111111 ECB mode from -> b'W\x88\\\xc7\x1c\x01\x00\xfe&q2\xa1\xebA\x0c\xb1'
```

```
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Catal\OneDrive\Desktop\proiectSI1>py KM.py
[RECEIVE->1]b'K1'
[RECEIVE->2]b'K1'
Sending K1 encrypted with K3 using ECB ->b'W\x88\\\xc7\x1c\x01\x00\xfe&q2\xa1\xebA\x0c\xb1'
```

A si B decripteaza cheia primita folosind K3

Dupa decriptarea cheilor , nodul B trimite un mesaj nodului A prin care il anunta ca poate incepe criptarea unui text cu K1/K2(in functie de modul de criptare).La primirea acestui mesaj , A va astepta pana cand se va introduce numele unui fisier din directorul tests existent (EX:test1) . Dupa introducerea numelui , A va incepe sa aplice modul de criptare ales pe acest text si va trimite pe rand cate un block de 16 bytes nodului B . Nodul B va decripta fiecare bloc si il va afisa. In caz ca numarul de bytes al textului nu este multiplu de 16 se va face padding .
$B \Longrightarrow server \Longrightarrow A$
$A \Longrightarrow server \Longrightarrow B$

Nodul A:

```
[RECEIVE->2]b'B-SEND-START'
Please enter the file from tests that you want to send : test1

===Sending to server the blocks encrypted with EBC and key-> 1111111111111111===
[ECB Block->1] b'\xb2\xaa|\xbd\xa9<\x14\xe0\xca\x1b#\x85\x05}\xdf\x0b'
[ECB Block->2] b'\x06\x14\x92\xf5%\x96{y\x15\xe19\xbbfOO\xaf'
[ECB Block->3] b'\xfc\xa0}\x8f\xb5\xc4\xf2\x8f\xced\x89\x1c\xcd\xa7\xc4_'
[ECB Block->4] b'\xe3\x82\xcc\xe3\x07\xcd\xee\xa7\xd3\x00\xf9\xc8\x17\x08\xd9+'
[ECB Block->5] b'\xea\xd9\x98\xe9\x0b0\x90\xd7\x10\xa2\xc4\x08R\xdf\x02i'
[ECB Block->6] b'\x84gy4\xd4\xc15\xd1/`\x19i\xc1\xc3\x89X'
[ECB Block->7] b'\xc5\xca\xee\xe18\xe8[\xc0c\xed\xb8\xc6.\xc1\xa9\xff'
[ECB Block->8] b'u2;\xbb\x0e\xf2\xb9\xae\xc9m\x95:xV\x92+'
[ECB Block->9] b'\xa13/X\xc4\xbeU\x8d\x00Kr?\xec\xd1k\xdd'
[ECB Block->10] b"\x03t:\x9c\xa7\x82\xe6>g\xcd\x8f'\x90\xc3\x1f\xb0"
[ECB Block->11] b'1\xb9\xf9\x9e\r~r,\xeaYeS$\x11K\xd1'
[ECB Block->12] b'jG\xc0h\x8f\xc47\x95\xc0\xb2\xa5\xde\xdd\xd2C\xd8'
[ECB Block->13] b'\xa3f\x17\x85XM\xc3\x9bf\x99\xf45\x0b\x86\xdc\xe6'
[ECB Block->14] b'`\xf5\xbf\r\xb5\xd0\xfe\x8a\xf0\x01K\t>E\x88\xd3'
[ECB Block->15] b'C\xc2\xad\x99\xafb\xc6\xae\x10n\x05\x99?KG\x9b'
[ECB Block->16] b'cN\xb5\xf3\xe7\xe8\xa0\xb5\xc1\xee|\xb00]o\xcc'
```

Nodul B:

```
============Setting decrypt class ECB and start decrypting the blocks with key 1111111111111111============
[ECB Block->1]b'\xb2\xaa|\xbd\xa9<\x14\xe0\xca\x1b#\x85\x05}\xdf\x0b' -> Lorem Ipsum is s
[ECB Block->2]b'\x06\x14\x92\xf5%\x96{y\x15\xe19\xbbfOO\xaf' -> imply dummy text
[ECB Block->3]b'\xfc\xa0}\x8f\xb5\xc4\xf2\x8f\xced\x89\x1c\xcd\xa7\xc4_' ->  of the printing
[ECB Block->4]b'\xe3\x82\xcc\xe3\x07\xcd\xee\xa7\xd3\x00\xf9\xc8\x17\x08\xd9+' ->  and typesetting
[ECB Block->5]b'\xea\xd9\x98\xe9\x0b0\x90\xd7\x10\xa2\xc4\x08R\xdf\x02i' ->  industry.
Lorem
[ECB Block->6]b'\x84gy4\xd4\xc15\xd1/`\x19i\xc1\xc3\x89X' ->  Ipsum has been
[ECB Block->7]b'\xc5\xca\xee\xe18\xe8[\xc0c\xed\xb8\xc6.\xc1\xa9\xff' -> the industry's s
[ECB Block->8]b'u2;\xbb\x0e\xf2\xb9\xae\xc9m\x95:xV\x92+' -> tandard dummy te
[ECB Block->9]b'\xa13/X\xc4\xbeU\x8d\x00Kr?\xec\xd1k\xdd' -> xt ever since th
[ECB Block->10]b"\x03t:\x9c\xa7\x82\xe6>g\xcd\x8f'\x90\xc3\x1f\xb0" -> e 1500s, when an
[ECB Block->11]b'1\xb9\xf9\x9e\r~r,\xeaYeS$\x11K\xd1' ->  unknown printer
[ECB Block->12]b'jG\xc0h\x8f\xc47\x95\xc0\xb2\xa5\xde\xdd\xd2C\xd8' ->  took a galley o
[ECB Block->13]b'\xa3f\x17\x85XM\xc3\x9bf\x99\xf45\x0b\x86\xdc\xe6' -> f type and scram
[ECB Block->14]b'`\xf5\xbf\r\xb5\xd0\xfe\x8a\xf0\x01K\t>E\x88\xd3' -> bled it to make
[ECB Block->15]b'C\xc2\xad\x99\xafb\xc6\xae\x10n\x05\x99?KG\x9b' -> a type specimen
[ECB Block->16]b'cN\xb5\xf3\xe7\xe8\xa0\xb5\xc1\xee|\xb00]o\xcc' -> book.


============ Decryption DONE ! ============
Lorem Ipsum is simply dummy text of the printing and typesetting industry.
Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
```

3 Testele efectuate pe diverse fisiere de intrare si observatiile efectuate

3.1 test1.txt

Aici am introdus mesaj mai lung , sunt 77 block-uri

3.2 test2.txt

Aici am introdus un mesaj care are mai putin de 16 bytes . Algoritmul va face padding pana la 16 bytes

3.3 test3.txt

Aici am introdus un mesaj mai ciudat care contine multe spatii si endline-uri.

3.4 test4.txt

Aici am introdus un mesaj care nu este multiplu de 16 ,el contine 20 bytes . Se va face padding pana la 32 bytes dupa care se va imparti in 2 block-uri , unul care contine primii 16 bytes din mesaj iar al doilea va avea pe primele 4 pozitii urmatorii 4 bytes din text . Pana la 16 se introduc valorile pozitiilor ramase de la padding(0X00).

4 Bibliografie

4.1 TCP server-client python

https://github.com/pricheal/python-client-server

4.2 Criptare cu AES

https://pycryptodome.readthedocs.io/en/latest/src/cipher/aes.html

4.3 ECB si OFB implementare

https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Output_feedback_(OFB)

4.4 Link cu proiect-ul pe github[PRIVAT MOMENTAN]

https://github.com/CozmaCatalin/proiectSI1