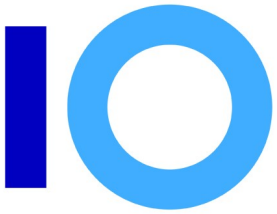


"Ss. Cyril and Methodius" University in Skopje
FACULTY OF COMPUTER
SCIENCE AND ENGINEERING

Лабораториска вежба бр. 4 по предметот

"Криптографија"

ECB, CBC and "cut-and-paste" cipher implementation

Изработил:

Мартин Костадинов

Број на индекс:

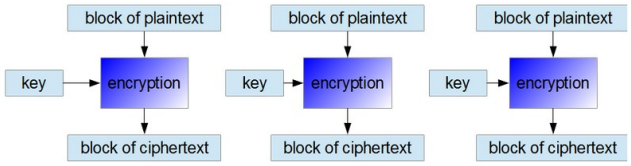
161159

ВОВЕД

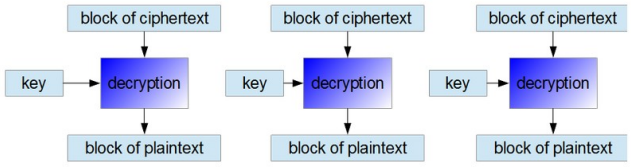
Во оваа документација ќе биде прикажан начинот на кој е имплементирана симулација за ECB и CBC модовите за работа како и симулација за cut-and-paste (исечи и залепи) нападот. ECB и CBC се модови за работа кои се често користени при блоковските алгоритми за криптирање и начинот на кој овие два мода функционираат е различен.

Кај ECB модот, пораката се дели на блокови од по 64 бита и секој блок се криптира со соодветен алгоритам за на крај да добиеме шифриран дел само од тој блок. Откако сите блокови ќе се шифрираат, тие се спојуваат во една порака и на овој начин се добива шифрираната порака.

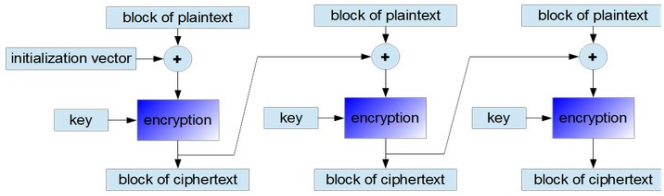
Кај CBC модот, пораката исто се дели на блокови од по 64 бита но овој мод е посложен и посигурен бидејќи во својата имплементација содржи иницијален вектор и секој шифриран дел од блокот влегува како вредност за иницијална вредност во наредниот блок.



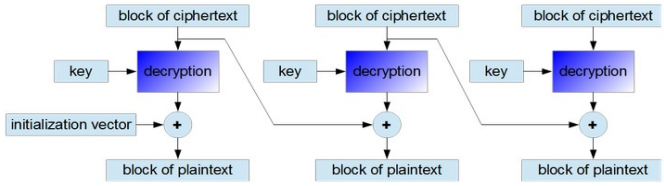
Encryption in the ECB mode



Decryption in the ECB mode



Encryption in the CBC mode



Decryption in the CBC mode

1. Имплементација

Оваа лабораториска вежба е потпомогнана од претходната (лабораториска вежба бр. 3) со тоа што е користен делот со DES алгоритмот, поточно делот за енкрипција и декрипција. Вежбата е правена во програмскиот јазик Java и во себе вклучува **5 класи**:

- **Block** (класа што претставува соодветна репрезентација на блок. Во неа имаме само една String променлива што ги претставува 64-те бита потребни за блокот);

- **BlockCBC** (класа што се користи за CBC модот на работа. Овде се сместени сите алгоритми за енкрипција, декрипција, замена на блокови и бришење на блокови);

- **BlockECB** (класа што се користи за ECB модот на работа, со сите потребни методи за декрипција/енкрипција, замена и бришење на блокови);

- **Utils** (помошна класа во која се сместени методите за цепкање на пораката во блокови од по 64 бита и сместување во хешмапа. Исто овде се сместени и методи за конвертирање од long во string како и од string во long);

- **Main** (главната класа каде што се тестира функционалноста на алгоритмите потребни за ECB и CBC модот како и нападот cut-and-paste).

2. Encryption and decryption

Во овој дел ќе биде претставена енкрипцијата и декрипцијата на овие модови на работа. Да претпоставиме дека за клуч ја имаме вредноста **591826312** (бинарно: 10001101000110100011011000100 0) и сакаме да ја енкриптираме пораката **Alice digs Bob. Trudy digs Tom.** При работа со ECB модот, ќе го добиеме следниот излез:

```
ECB MODE:
;N□´□$§Î|2□³øyO□tA+□□ T6dÿiÔû□.ù
------------------------------------------------------------------------
Alice digs Bob. Trudy digs Tom.
```

Ако сакаме да го користиме CBC модот, потребно е да дефинираме и иницијален вектор. Во случајот иницијалниот вектор е **0x00003442A91077DE45AC** во хексадецимален запис. Излезот при CBC е следниот:

```
CBC MODE:
□ól□èóU¾x□Ç_PÇ  □Ó^□□ýÏ□$ðxþ +^
------------------------------------------------------------------------
Alice digs Bob. Trudy digs Tom.
```

3. cut-and-paste

За успешна имплементација на овој напад потребно е да дефинираме методи што ќе вршат замена или бришење на блоковите. Методата е достапна во делот со Java кодот кај двата модови на работа.

```
void replaceBlocks(int block_1, int block_2) {
    long block1 = cipherMap.get(block_1);
    long block2 = cipherMap.get(block_2);

    cipherMap.put(block_1, block2);
    cipherMap.put(block_2, block1);
}

public void deleteBlock(int block_number) {
    cipherMap.remove(block_number);
}
```

При користење на истиот текст (**Alice digs Bob. Trudy digs Tom.**), текстот се дели на 4 блока од по 8 карактери т.е вкупно 64 бита по блок. Ако при ECB мод на работа ги смениме третиот и првиот блок, т.е ја повикаме методата **blockECB.replaceBlocks(3,1)**, нападот ќе се изврши успешно т.е ќе добиеме неточна реалност:

```
ECB MODE:
;N□´□$§Î|2□³øyO□ŁA+□□ T6dÿìÔû□.ù
--------------------------------------------------------------------------------
Alice digs Tom. Trudy digs Bob.
```

Кај CBC модот ова не функционира поради тоа што секој блок (освен првиот) зависи од претходниот и смена во еден блок би значело смена во сите наредни блокови, почнувајќи од сменетиот+1.

```
CBC MODE:
□ól□èóU¾✗□Ç_PÇ  □Ó^□□ýÏ□$ðx þ +^
--------------------------------------------------------------------------------
Alice di♯□□u□□□Q┬┤@BCYlG/♯□□c□□□Q
```

Бришењето на блокови е друг начин кој се користи при овој вид на напад. При ECB модот на работа и бришењето на првиот (1) и вториот (2) блок користејќи ја истата порака, се добива следниот израз:

```
ECB MODE:
;N□´□$§Î|2□³øyO□ŁA+□□ T6dÿìÔû□.ù
--------------------------------------------------------------------------------
Alice digs Tom.
```

Kaj CBC модот повторно не добиваме ништо позитивно од иста причина како и при замена на блоковите.