Cryptography Coursework 3 - AES

ricardo.moreira@acad.pucrs.br

May 19, 2015

Abstract

This paper gives a very short introduction to Advanced Encryption Standard (AES) and describes its usage for encrypting and decrypting samples in the context of Cryptography course classwork nº3.

1 Introduction

The Advanced Encryption Standard was the result of a process for selecting a substitution to Data Encryption Standard (DES) that was notable for its openness and its international flavor [[1]]. There were 21 candidates submitted from a variety of countries and after evaluated according to security, cost and algorithm implementation characteristics, AES was adopted as a standard on November 26, 2001.

2 AES with PyCrypto

The PyCrypto library provides interface to variety of encryption algorithms, the central goal is to provide a consistent interface to make it extremely easy to replace old algorithms with newer, more secure ones.

The code required to implement the encryption and decryption methods are really short, and an experienced Python developer could make them even more concise that the code below:

```
def aes_decrypt(key, mode, iv, cipher):
    if mode == AES.MODE_CTR:
        ctr = Crypto.Util.Counter.new(128,
                initial_value = long(iv.encode("hex"), 16))
        aes = AES.new(key, mode, counter = ctr)
    else:
        aes = AES.new(key, mode, iv)

    return aes.decrypt(cipher)
```

```
def aes_encrypt(key, mode, plain):
    if mode == AES.MODE_CTR:
        ctr = Crypto.Util.Counter.new(128,
                initial_value = long(iv.encode("hex"), 16))
        aes = AES.new(key, mode, counter = ctr)
    else:
        aes = AES.new(key, mode, iv)

    return aes.encrypt(plain)
```

3 Modes of operation

A block cipher by itself is only a secure transformation of one fixed-length block. The modes of operation describe how to repeatedly apply the cipher's single-block operation to securely transform amounts of data larger than a block.

We have used two modes of operation. The first one is Cipher Block Chaining (CBC mode) which is the most commonly used mode of operation. In CBC, each block of plaintext is XORed with the previous ciphertext block before encryption, making the encryption a sequential process. To make each message unique, an initialization vector must be used in the first block. CBC encryption and decryption formulas:

$$C_i = E_k(P_i \oplus C_{i-1}), C_0 = IV \tag{1}$$

$$P_i = D_k(C_i) \oplus C_{i-1}, C_0 = IV \tag{2}$$

The second mode of operation we used is Counter (CTR mode). The CTR mode turns a block cipher into a stream cipher by encrypting successive values of a "counter". The counter can be any function that produces a sequence that is guaranteed not to repeat for a long time. CTR mode is well suited to operate on a multi-processor machine where blocks can be encrypted in parallel.

4 Padding

The primary use of padding with classical ciphers is to prevent the cryptanalyst from using that predictability to find cribs that aid in breaking the encryption. ECB mode require plaintext input that is a multiple of the block size, so messages have to be padded to bring them to this length.

The padding scheme used is the PKCS5, the value of each added byte is the number of bytes that are added. The padding will be one of:

```
01
02 02
03 03 03
04 04 04 04
```

```
05 05 05 05 05
etc
```

References

[1] Cryptography: Theory and Practice, Stinson, Douglas R., CRC Press, Inc., 1995.

A

Coursework 3 tasks and results

```
=> task 1
 decrypt mode CBC key 140b41b22a29beb4061bda66b6747e14
 buffer:
 4ca00ff4c898d61e1edbf1800618fb2828a226d160dad07883d04e
 008a7897ee2e4b7465d5290d0c0e6c6822236e1daafb94ffe0c5da
 05d9476be028ad7c1d81

 decrypted: Basic CBC mode encryption needs padding.
 hex:
 4261736963204342432064f6465206520656e6372797074696f6e206e
 6565647320706164646965672e0808080808080808

=> task 2
 decrypt mode CBC key 140b41b22a29beb4061bda66b6747e14
 buffer:
 5b68629feb8606f9a6667670b75b38a5b4832d0f26e1ab7da33249
 de7d4afc48e713ac646ace36e872ad5fb8a512428a6e21364b0c37
 4df45503473c5242a253

 decrypted: Our implementation uses rand. IV
 hex:
 4f757220696d706c656d656e746174696f6e20757365732072616e
 642e204956101010101010101010101010101010

=> task 3
 decrypt mode CTR key 36f18357be4dbd77f050515c73fcf9f2
 buffer:
 69dda8455c7dd4254bf353b773304eec0ec7702330098ce7f7520d
 1cbbb20fc388d1b0adb5054dbd7370849dbf0b88d393f252e764f1
 f5f7ad97ef79d59ce29f5f51eeca32eabedd9afa9329

 decrypted: CTR mode lets you build a stream cipher from a block cipher.
 hex:
 435452206d6f6465206c65747320796f75206275696c6420612073
 747265616d2063697068657220666726f6d206120626c6f636b2063
 69706865722e
```

```
=> task 4
 decrypt mode CTR key 36f18357be4dbd77f050515c73fcf9f2
 buffer:
 770b80259ec33beb2561358a9f2dc617e46218c0a53cbeca695ae4
 5faa8952aa0e311bde9d4e01726d3184c34451

 decrypted: Always avoid the two time pad!
 hex: 416c776179732061766f69642074686520747776f2074696d652070616421

=> task 5
 encrypt mode CTR key 36f18357be4dbd77f050515c73fcf9f2
 buffer:
 5468697320697320612073656e74656e636520746f20626520656e
 6372797074656420757369e672041455320616e6420435452206d6f64652e

 plain: This is a sentence to be encrypted using AES and CTR mode.
 encrypted:
 f16606d2fc26ed8b7e15fe5ee4895fa14d204cc5d21a0a7a28749ac15209e5
 c01667812867bb694916a956d6e9c09f3fbb355485162b20e2d9e3

=> task 6
 encrypt mode CBC key 140b41b22a29beb4061bda66b6747e14
 buffer:
 4e6578742054687572736461792 06f6e65206f662074686520626573742074
 65616d7320696e2074686520776f726c642077696c6c206661636520612062
 6967206368616c6c656e676520696e20746865204c696265727461646f7265
 7320646120416d65726963612043686 16d70696f6e736869702e

 plain:
 Next Thursday one of the best teams in the world will face a big challenge in
 the Libertadores da America Championship.XXXXXXXXX

 encrypted:
 9ad6561db984aa94a87388438a1674cb7b5125b2ef3e44c2bd9b580be76857
 085c929e5658635f0a75cc3ac96ebd309db20258251e8641230048e71ba17f
 80f7d23032f704a88c9ab504d15be2f63f72ad76f289877858b9617137612c
 40d3dd312f74ab12a95e33c2ab3240cd3718982553c132206fa40dcb95ea7f
 c6e71e39
```