ISSN: 2754-6705

Journal of Mathematical & Computer Applications

SCIENTIFIC Research and Community

Review Article

Open Access

Cybersecurity: Fast Encryption Cipher Block Chaining Mode (FCBC Mode) for Time Series Data

Binoy Kurikaparambil Revi

Independent Research, USA

ABSTRACT

Cipher Block Chaining (CBC) mode is used to make the encryption of each block of data in a data series more secure by making the encryption of the block interdependent. In case of time series data, the size of each data point includes the time information and actual data itself. This means that each block of data that is encrypted may hold only nearly half the actual data values due to timestamp information. This reduces the efficiency of encryption and decryption of data for applications that demand high frequency data. Fast Encryption Cipher Block Chaining Mode (FCBC Mode) provides a solution to overcome this to a great extent.

***Corresponding author**

Binoy Kurikaparambil Revi, Independent Research, USA.

Received: April 05, 2024; **Accepted:** April 15, 2024, **Published:** April 25, 2024

Introduction

Time series data or in other words data that have a timestamp associated with it, has become a very common data format in the modern world due to the emergence of Internet of Things(IoT), Data Mining and Analytics. In most of the real world applications, these data can have direct or indirect implications in the form of responses or actions. This brings the cybersecurity question on to the table, Why if someone can act on this data to manipulate the data or hinder the data to alter or suppress responses and actions. Answer is yes!

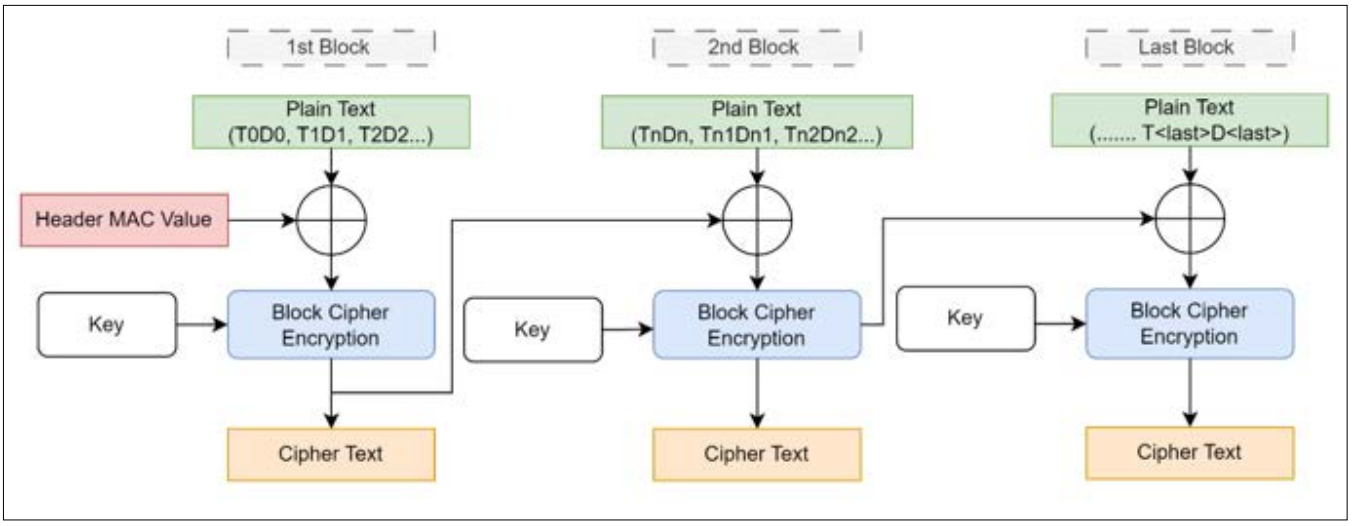
The simplest way to solve this is to encrypt the data at the source and then transmit to the destination. There are various algorithms and strategies to do the encryption of these data to provide the level of security needed. Sensitivity of the data and efficiency are the two major factors that control the decision of which algorithm and architecture to use to attain that level of security expected.

Serializable and Parallelizable - Encryption and Decryption

Typically a data stream can be chunked to blocks to perform the encryption. These blocks can be processed in parallel or in serial mode depending on the encryption algorithm that is used. Parallelizable encryption and decryption of data blocks bring efficiency to the process while some of the strong block encryption techniques like the Cipher Block Chaining(CBC) Mode uses Serial encryption.

Cipher Block Chaining (CBC)

The advantage of using Cipher Block Chaining Mode is that even though the encryption uses serial mode, the decryption is parallelizable. The encryption is not parallelizable because encryption of the block has interdependability. For encrypting each block, the algorithm receives key, cipher block from previous encrypted block and plain text to create corresponding ciphertext block.



As we can see in the figure above, the plain text contains the time series data with a series of pairs, where the first value of the pair is the timestamp and the second value of the pair is a single data or a vector of multiple data.

Now let us consider the usability of these data in the applications. From the analysis done on various applications, the data is used mainly for the scenarios mentioned below:

- **Data Monitoring:** In this case, the incoming data stream is monitored to make sure that the data values fall within a predefined range or under certain criteria or set of rules. Here the precision of the data is more important and the timestamp associated with it acts as an input to monitor the data trends. In many applications, the data is pre-conditioned before being used by such an application by different methods such as interpolation, data filtering etc…
- **Data Storage:** For data storage, preprocessing data seems to be a very good option before the data is actually stored. This is because the data captured to use by application is more specific than generic. So if the data is stored in raw format, everytime the application reads the data, it has to preprocess the data. In case of the time series data, there may be a possibility of increasing retention period of data if the data is preprocessed and stored.

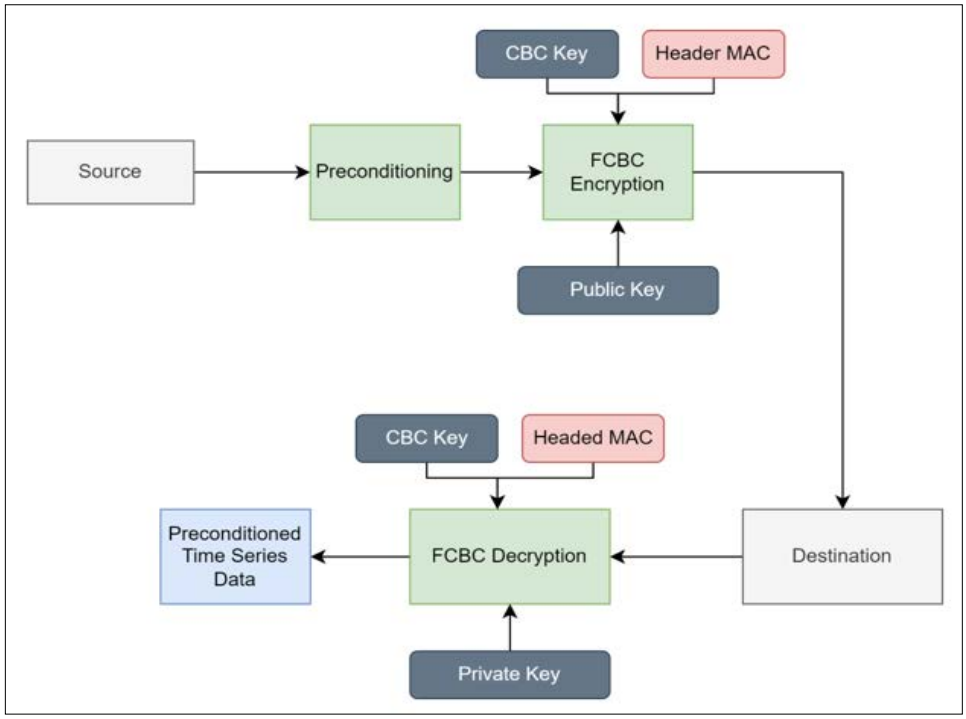
After the analysis of these scenarios, it is understood that in majority of the cases, prep-conditioning of data along with the data processing and filtering mechanisms can very well bring the efficiency of the application to a great extent.

Predictability of the Time

Let us analyze the preconditioned data that is going to be encrypted before it is transmitted or before it gets stored. Classic way to do this is to encrypt each block of data before the transmission. However the biggest question is, Do we need to encrypt the time stamp? Why can't we predict the time if we know the parameters from preconditioning the data? Analysis shows that in the majority of cases where preconditioned data is used, the timestamp can be calculated from the parameters used to precondition the data. Fast Encryption Cipher Block Chaining Mode (FCBC Mode) provides a solution to encrypt the time series data without encrypting the timestamp.

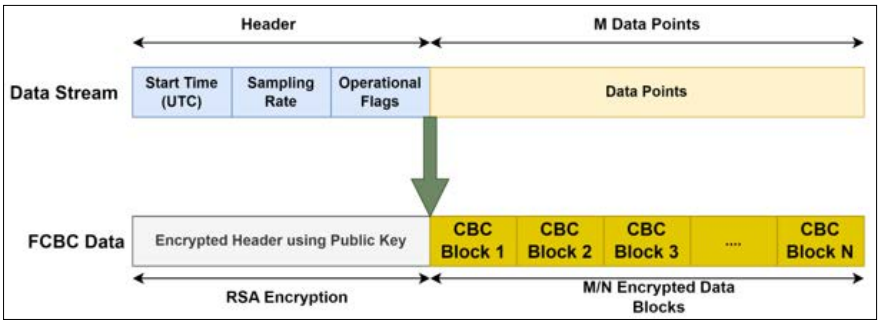
Fast Encryption Cipher Block Chaining Mode (FCBC Mode)

Fast Encryption Cipher Block Chaining Mode is used to encrypt and decrypt pre-conditioned time series data series. The following algorithm describes the encryption and decryption of the data series using FCBC mode.

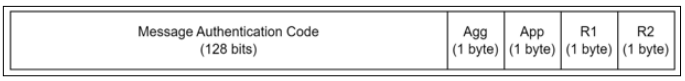


Step1: Create Header

Use the preconditioning parameter to create the header for the preconditioned data series. The deader contain 3 sections:



- **Start time [64 bits]:** This is the start time of the sample data in UTC
- **Sampling Rate [64 Bits]:** This is the sampling rate of the data. Note that this is not the sampling rate of the actual data. This is the sampling rate of the preconditioned data.
- **Operation Flags[160 Bits]:** This field contains 5 pieces of information where the first one is Message Authentication code and remaining are 4 single byte flags.



Message Authentication Code(MAC) : This is the MAC value computed from Start Time, Sampling Rate, Agg Value, App Value, R1 and R2.

Agg Value: The following codes are used for the aggregator notations:

Aggregator code(in Hex Format)	Description
0X00	Total number of values
0X01	Mean of values
0X02	Sum of values
0X04	Minimum value
0X03	Maximum value
0X05	Difference between maximum and minimum value
0X06	First Value
0X07	Last Value
0X08	Population standard deviation of values
0X09	Sample standard deviation of values
0X0A	Population variance of the values
0X0B	Sample variance of the values
0X0C	Time-weighted average over time frame
0XFF	Error

App: This is the application code that is custom to the application.

R1: This is used to indicate the data source

R2: Reserved for optional use

Step3: Encrypt Header

After creating the header, the header is encrypted using Asymmetric encryption. This encryption is done at the data source using RSA. As the encryption is symmetric and uses a public key it helps to support multiple data sources very easily. The header can be only decrypted by the destination application using the private key. Note that the encryption is done using the complete header including MAC value. Header encryption is just done once and is transmitted at the start of the data transmission.

Step3: FCBC Mode Encryption

Data packets are encrypted using Cipher Block Chaining Mode. In FCBC mode, before the encryption, the timestamp information is removed from the time series data. The main difference between the classic CBC mode encryption and FCBC mode encryption is that FCBC mode doesn't use any initialization vector (IV). Instead, it uses the MAC value embedded in the header prior to header encryption as the IV. This way if the data stream needs to be decrypted, first the header must be decrypted.

Step5: Data Transmission

The data transmission mode is determined by the application considering the sensitivity of the data, industry standards and protocols.

Step5: FCBC Mode Decryption

In this step, the data received by the destination is decrypted. The header is first decrypted using a private key with the RSA decryption technique. After decrypting the header, the header MAC is calculated and validated against the MAC stored in the header. On confirming the data integrity, the MAC value is used as IV to decrypt the data blocks using Cipher Block Chaining mode.

Advantages of Using the Fast Encryption Cipher Block Chaining Mode (FCBC Mode)

- A major benefit in using the FCBC mode is that the amount of data that is encrypted and transmitted is much less as the timestamp information is stripped off the data series.
- Compared to classic CBC mode, FCBC mode doesn't use IV to start block encryption. Instead, the MAC value is dependent on multiple data and importantly start time in UTC that can be considered as a randomly unique number.
- Asymmetric encryption offers a high level of security, and the speed is not impacted as it is used only to encrypt the header. This also helps multiple data sources to perform the encryption using the public key.
- Even if the data is preconditioned, the destination gets all information on the precondition parameters from the header.

References

1. Samanta D, Alahmadi AH, Karthikeyan MP, Khan MZ, Banerjee A, et al. (2021) Cipher Block Chaining Support Vector Machine for Secured Decentralized Cloud Enabled Intelligent IoT Architecture. in IEEE Access 9: 98013-98025.
2. Sinurat S, Pasaribu M (2021) Text Encoding Using Cipher Block Chaining Algorithm. Info Sains 11: 13-17.
3. Huang YL, Yie Leu F, Liu JC, Yang JH, Yu CW, et al. (2013) Building a block cipher mode of operation with feedback keys. 2013 IEEE International Symposium on Industrial Electronics, Taipei, Taiwan 1-4.
4. Christensen C (2010) Review of Cryptography and Network Security: Principles and Practice, Fifth Edition Cryptologia 35: 97-99.