A Critical Assessment on Block Cipher Modes
Satvika¹, Akhil Kaushik²
Assistant Professor¹, ²
Department of IT¹, Department of CSE²
T.I.T College, Bhiwani, India

Abstract:
The technological rise has given an exponential ascended in the security issues of the crucial data. Usually the data that is shared on the internet now-a-days is done through social media sites like Facebook, twitter, etc. This data in non-important in nature and hence is of no use to the hackers. On the other hand, financial data or personal data is a great lure for the eavesdroppers and hackers. Thus depending upon the data, the security must be implemented for safeguarding of such vital information. A lot of techniques have been deployed for protection of data, one of which is block encryption of data. This paper discusses some vital modes of block cipher and compares them to find the best security alternative.

Index Terms: Block Cipher, ECB Mode, CBC Mode, CFB Mode, OFB Mode, PCBC Mode, CTR Mode.

I. INTRODUCTION
With the continual expansion of multimedia and Internet applications, the needs and requirements of the technologies used, grew and evolved. In cryptography[1][8], a block cipher[2] is a deterministic algorithm operating on fixed-length groups of bits, called blocks, with an unvarying transformation that is specified by a symmetric key. Block ciphers are important elementary components in the design of many cryptographic protocols[3], and are widely used to implement encryption of bulk data. The modern design of block ciphers is based on the concept of an iterated product cipher. Product ciphers were suggested and analyzed by Claude Shannon in his seminal 1949 publication Communication Theory of Secrecy Systems as a means to effectively improve security by combining simple operations such as substitutions and permutations. Iterated product ciphers carry out encryption in multiple rounds, each which uses a different sub-key derived from the original key. One widespread implementation of such ciphers is called a Feistel network, named after Horst Feistel, and notably implemented in the DES[4] cipher. Many other realizations of block ciphers, such as the AES[8], are classified as substitution-permutation networks. Block ciphers are not secure encryption schemes. Rather, they are building blocks that can be used to construct secure schemes. Today, there is a palette of attack techniques against which a block cipher must be secure, in addition to being robust against brute force attacks. Even a secure block cipher is suitable only for the encryption of a single block under a fixed key. A multitude of modes of operation have been designed to allow their repeated use in a secure way, commonly to achieve the security goals of confidentiality and authenticity. However, block ciphers may also be used as building blocks in other cryptographic protocols, such as universal hash functions and pseudo-random number generators.

With well-established encryption algorithms like DES or AES at hand, one could have the impression that most of the work for building a cryptosystem - for example a suite of algorithms for the transmission of encrypted data over the internet - is already done. But the task of a cipher is very specific: to encrypt or decrypt a data block of a specified length. Given a plaintext of arbitrary length, the simplest approach would be to break it down to blocks of the desired length and to use padding for the final block. Each block is encrypted separately with the same key, which results in identical ciphertext blocks for identical plaintext blocks.

II. MODES OF BLOCK CIPHER
A mode of operation is essentially a way of encrypting arbitrary-length messages using a block cipher (i.e., pseudorandom permutation)[4][8]. The messages can be unambiguously padded to a total length that is a multiple of the block size by appending a 1 followed by sufficiently-may 0’s. In cryptography, a mode of operation is an algorithm that uses a block cipher to provide an information service such as confidentiality or authenticity. A block cipher by itself is only suitable for the secure cryptographic transformation (encryption or decryption) of one fixed-length group of bits called a block. A mode of operation describes how to repeatedly apply a cipher's single-block operation to securely transform amounts of data larger than a block. Most modes require a unique binary sequence, often called an initialization vector (IV)[4], for each encryption operation. The IV has to be non-repeating and for some modes random as well. The initialization vector is used to ensure distinct ciphertexts are produced even when the same plaintext is encrypted multiple times independently with the same key. Block ciphers have one or more block size(s), but during transformation the block size is always fixed. Block cipher modes operate on whole blocks and require that the last part of the data be padded to a full block if it is smaller than the current block size. There are, however, modes that do not require padding because they effectively use a block cipher as a stream cipher. Historically, encryption modes have been studied extensively in regard to their error propagation properties under various scenarios of data modification. Later development regarded integrity protection as an entirely separate cryptographic goal. Some modern modes of operation combine

Figure 1. Working of Block Cipher
confidentiality and authenticity in an efficient way, and are known as authenticated encryption modes.

III. ELECTRONIC CODE BOOK MODE

Given a plaintext of arbitrary length, the simplest approach would be to break it down to blocks of the desired length and to use padding for the final block. Each block is encrypted separately with the same key, which results in identical cipher text blocks for identical plaintext blocks. This is known as Electronic Code Book (ECB) mode of operation, and is not recommended in many situations because it does not hide data patterns well.

If the first block has index 1, the mathematical formula for CBC encryption is

\[ C_i = E_K(P_i \oplus C_{i-1}), C_0 = IV \]

While the mathematical formula for CBC decryption is

\[ P_i = D_K(C_i) \oplus C_{i-1}, C_0 = IV. \]

CBC has been the most commonly used mode of operation. Its main drawbacks are that encryption is sequential (i.e., it cannot be parallelized), and that the message must be padded to a multiple of the cipher block size. One way to handle this last issue is through the method known as ciphertext stealing. Note that a one-bit change in a plaintext or IV affects all following ciphertext blocks.

V. PROPAGATING CIPHER-BLOCK CHAINING

The Propagating Cipher-Block Chaining (PCBC) or Plaintext Cipher-Block Chaining mode [7] was designed to cause small changes in the ciphertext to propagate indefinitely when decrypting, as well as when encrypting.

Encryption and decryption algorithms are as follows:

\[ C_i = E_K(P_i \oplus P_{i-1} \oplus C_{i-1}), P_0 \oplus C_0 = IV \]

\[ P_i = D_K(C_i) \oplus P_{i-1} \oplus C_{i-1}, P_0 \oplus C_0 = IV. \]

PCBC is used in Kerberos v4 and WASTE, most notably, but otherwise is not common. On a message encrypted in PCBC mode, if two adjacent ciphertext blocks are exchanged, this does not affect the decryption of subsequent blocks. For this reason, PCBC is not used in Kerberos v5.

VI. CIPHER FEEDBACK

The Cipher Feedback (CFB) [8] [3] mode, a close relative of CBC, makes a block cipher into a self-synchronizing stream cipher. Operation is very similar; in particular, CFB decryption is almost identical to CBC encryption performed in reverse. This simplest way of using CFB described above is not any more self-synchronizing than other cipher modes like CBC. If a whole block size of ciphertext is lost both CBC and CFB will synchronize, but losing only a single byte or bit will permanently throw off decryption. To be able to synchronize after the loss of only a single byte or bit, a single byte or bit must be encrypted at a time. CFB can be used this way when combined with a shift register as the input for the block cipher.

\[ C_i = E_K(C_{i-1} \oplus P_i), P_i = E_K(C_{i-1} \oplus C_i) \]
\[ C_0 = IV \]

\[ C_j = P_j \oplus O_j \]
\[ P_j = C_j \oplus O_j \]
\[ O_j = E_K(I_j) \]
\[ I_j = O_{j-1} \]
\[ I_0 = IV \]

Figure 6. Block Cipher in CFB Mode

To use CFB to make a self-synchronizing stream cipher that will synchronize for any multiple of x bits lost, start by initializing a shift register the size of the block size with the initialization vector. This is encrypted with the block cipher, and the highest x bits of the result are XORed with x bits of the plaintext to produce x bits of ciphertext. These x bits of output are shifted into the shift register, and the process repeats with the next x bits of plaintext. Decryption is similar, start with the initialization vector, encrypt, and XOR the high bits of the result with x bits of the ciphertext to produce x bits of plaintext. Then shift the x bits of the ciphertext into the shift register. This way of proceeding is known as CFB-8 or CFB-1 (according to the size of the shifting). CFB shares two advantages over CBC mode with the stream cipher modes OFB and CTR: the block cipher is only ever used in the encrypting direction, and the message does not need to be padded to a multiple of the cipher block size (though ciphertext stealing can also be used to make padding unnecessary).

VII. OUTPUT FEEDBACK MODE

The Output Feedback (OFB)[8][2] mode makes a block cipher into a synchronous stream cipher. It generates keystream blocks, which are then XORed with the plaintext blocks to get the ciphertext. Just as with other stream ciphers, flipping a bit in the ciphertext produces a flipped bit in the plaintext at the same location. This property allows many error correcting codes to function normally even when applied before encryption. Because of the symmetry of the XOR operation, encryption and decryption are exactly the same:

\[ C_j = P_j \oplus O_j \]
\[ P_j = C_j \oplus O_j \]
\[ O_j = E_K(I_j) \]
\[ I_j = O_{j-1} \]
\[ I_0 = IV \]

Figure 7. Block Cipher in OFB Mode

Each output feedback block cipher operation depends on all previous ones, and so cannot be performed in parallel. However, because the plaintext or ciphertext is only used for the final XOR, the block cipher operations may be performed in advance, allowing the final step to be performed in parallel once the plaintext or ciphertext is available. It is possible to obtain an OFB mode key stream by using CBC mode with a constant string of zeroes as input. This can be useful, because it allows the usage of fast hardware implementations of CBC mode for OFB mode encryption. Using OFB mode with a partial block as feedback like CFB mode reduces the average cycle length by a factor of 2^x or more. A mathematical model proposed by Davies and Parkin and substantiated by experimental results showed that only with full feedback an average cycle length near to the obtainable maximum can be achieved. For this reason, support for truncated feedback was removed from the specification of OFB.[8]

VIII. COUNTER MODE

Like OFB[1], counter mode turns a block cipher into a stream cipher. It generates the next keystream block by encrypting successive values of a counter. The counter can be any function which produces a sequence which is guaranteed not to repeat for a long time, although an actual increment-by-one counter is the simplest and most popular. The usage of a simple deterministic input function used to be controversial; critics argued that "deliberately exposing a cryptosystem to a known systematic input represents an unnecessary risk." By now, CTR mode is widely accepted, and problems resulting from the input function are recognized as a weakness of the underlying block cipher instead of the CTR mode. Along with CBC, CTR mode is one of two block cipher modes recommended by Niels Ferguson and Bruce Schneier. CTR mode has similar characteristics to OFB, but also allows a random access property during decryption. CTR mode is well suited to operate on a multi-processor machine where blocks can be encrypted in parallel. Furthermore, it does not suffer from the short-cycle problem that can affect OFB. Note that the nonce in this graph is the same thing as the initialization vector (IV) in the other graphs. The IV/nonce and the counter can be combined together using any lossless operation (concatenation, addition, or XOR) to produce the actual unique counter block for encryption.

Figure 8. Block Cipher in Counter Mode

IX. CONCLUSION

This paper focuses on many different approaches of encrypting data and suggests that the fastest alternate of data encoding data is encrypting data using block ciphers. Nevertheless there have been modes possible through which any block cipher can be implemented. Every working mode of block cipher has its own advantages and disadvantages, hence the usage of various block...
cipher modes depend on the working situation. Where the OFB mode is best for auto-generation of keys, the CBC mode is suited where plaintext needs to be altered before encryption. However, PCBC mode stands out as the best choice as it alters not only crypto-key at various rounds, but also the plaintext. This particular feature of PCBC mode safeguards data against plaintext attacks and replay attacks. Thus it can be concluded that PCBC mode is the most secure approach of all working modes of block cipher.

X. REFERENCES


[6]. http://www.iks-jena.de/ mitarb/lutz/security/ cryptfaq/q84.html