Block Cipher Modes of Operation

Luke Anderson

18th March 2016

University Of Sydney

Overview

Crypto-Bulletin

From Stolen Wallet to ID Theft, Wrongful Arrest
http://krebsonsecurity.com/2016/03/from-stolen-wallet-to-id-theft-wrongful-arrest/

Google doubles reward for security bug hunters
http://www.itnews.com.au/news/google-doubles-reward-for-security-bug-hunters-416879

Anti-DDoS firm Staminus ransacked by hackers
http://www.itnews.com.au/news/anti-ddos-firm-staminus-ransacked-by-hackers-416834

Slew of dangerous Adobe Flash flaws patched
Remote code execution vulnerabilities galore.

http://www.itnews.com.au/news/slew-of-dangerous-adobe-flash-flaws-patched-416771

Modes Of Operation

Cipher Modes of Operation

Block ciphers by themselves only encrypt a single block of data.

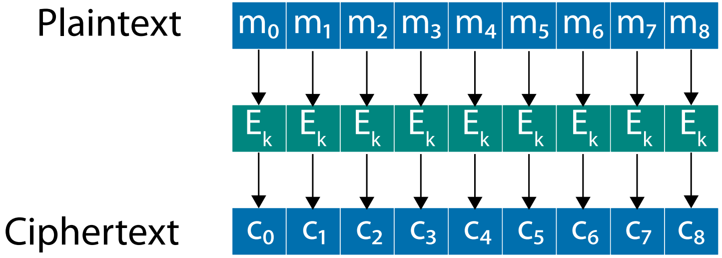
By using different modes of operation, messages of an arbitrary length can be split into blocks and encrypted using a block cipher. Each mode of operation describes how a block cipher is repeatedly

applied to encrypt a message and has certain advantages and disadvantages.

Electronic Code Book (ECB)

Electronic Code Book (ECB) encrypts each block separately.

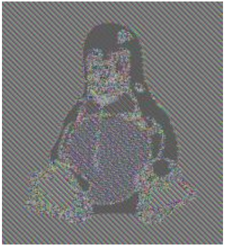
ECB is generally an insecure and naïve implementation, it is vulnerable to a range of attacks; including dictionary and frequency attacks.



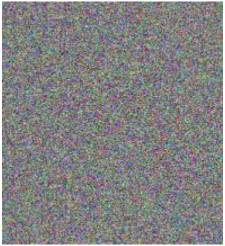
Electronic Code Book (ECB)

The problem with ECB:

(a) Original Image



(b) ECB mode



(c) Other mode

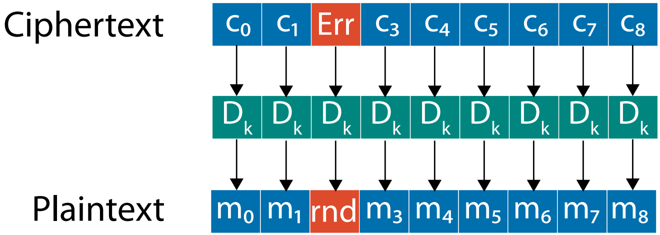
Encryption of Tux[1] image.

[1] Tux is the Linux mascot

ECB Properties

Identical plaintext blocks result in identical ciphertext blocks
Since blocks are enciphered independently, a reordering of ciphertext blocks results in reordering of plaintext blocks.
ECB is thus not recommended for messages ¿ 1 block in length.

Error propagation: Bit errors only impact the decoding of the corrupted block (block will result in gibberish)

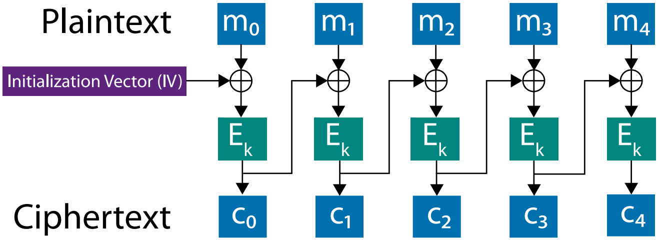


Error propagation in ECB

Cipher Block Chaining (CBC)

In **Cipher Block Chaining (CBC)** blocks are chained together using XOR.

The **Initialisation Vector (IV)** is a random value that is transmitted in the clear that ensures the same plaintext and key does not produce the same ciphertext.



CBC Mode Encryption

CBC Properties

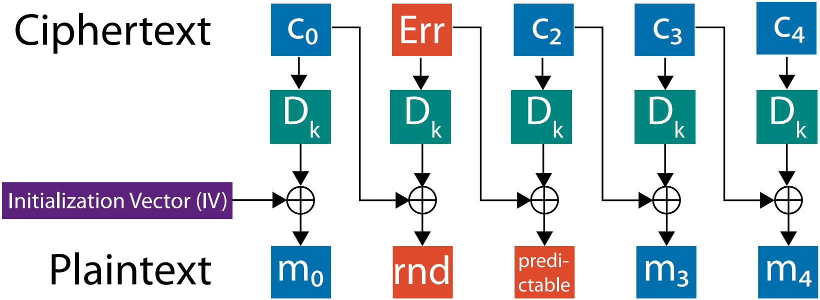
Identical plaintexts result in identical ciphertexts when the same plaintext is enciphered using the same key and IV.
Changing at least one of $[k, IV, m_0]$ affects this.

Rearrangement of ciphertext blocks affects decryption, as ciphertext part c_j depends on all of $[m_0, m_1, \cdots, m_j]$.

Error propagation:
Bit error in ciphertext c_j affects deciphering of c_j and c_{j+1}.Recovered block m'_j typically results in random bits.
Bit errors in recovered block m'_{j+1} are precisely where c_j was in error. Attacker can cause predictable bit changes in m_{j+1} by altering c_j.

Bit recovery:
CBC is self-synchronising in that if a bit error occurs in c_j but not c_{j+1}, then c_{j+2} correctly decrypts to m_{j+2}.

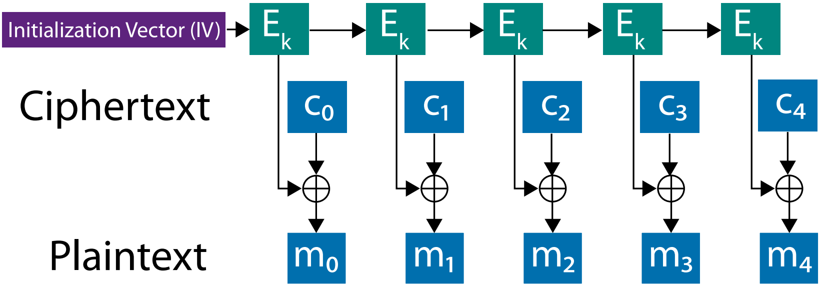
CBC Error Recovery



CBC Decryption: Ciphertext errors only affect two plaintext blocks, one in a predictable way.

Output Feedback Mode (OFB)

Output Feedback Mode (OFB) effectively turns a block cipher into a synchronous stream cipher.



OFB Properties

Identical plaintext results in identical ciphertext when the same plaintext is enciphered using the same key and IV.

Chaining Dependencies: The key stream is plaintext independent.

Error propagation: Bit errors in ciphertext blocks cause errors in the same position in the plaintext.

Error recovery: Recovers from bit errors, but not bit loss (misalignment of key stream)

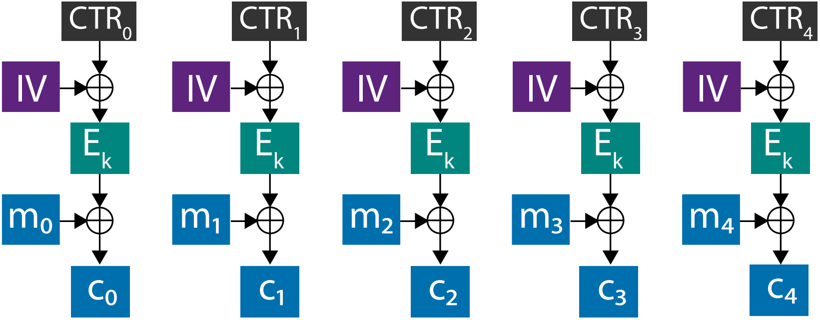
Throughput: Key stream may be calculated independently (e.g. pre-computed)

IV must change: Otherwise it becomes a two time pad.

Counter Mode (CTR)

Counter Mode (CTR) modifies the IV for each block using a predictable counter function.

The counter can be any function (e.g. a PRNG), but it is commonly just an incrementing integer.



CTR Mode Encryption

Evaluating Block Ciphers & Modes

Estimated Security Level:
Confidence grows the more it is analysed.

Key Size:
Upper bound on security, but longer keys add costs (generation, storage, etc.)

Throughput:
How fast can it be encrypted/decrypted?
Can it be pre-computed?

Block Size:
Larger is better to reduce overheads, but is more costly.

Data Expansion:
Ciphertext may be much larger than plaintext.

Error Propagation:
What happens as a result of bit errors or bit loss?