MH4311 Cryptography

Lecture 7
Block Cipher
Part 4, Modes of Operation

Wu Hongjun

Lecture Outline

- Classical ciphers
- Symmetric key encryption
  - One-time pad & information theory
  - Block cipher
    - DES, Double DES, Triple DES
    - AES
    - Modes of Operation
    - Attacks
  - Stream cipher
- Hash function and Message Authentication Code
- Public key encryption
- Digital signature
- Key establishment and management
- Introduction to other cryptographic topics

Recommended Reading

- CTP Section 3.7
- HAC Section 7.2.2
- Wikipedia:
  - Modes of operation

    http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation
  - Ciphertext Stealing

    http://en.wikipedia.org/wiki/Ciphertext_stealing

Block cipher

- **Fixed block size**
  - **DES: 64 bits**
  - **AES: 128 bits**
- **How to encrypt a long message?**
- **How to encrypt many messages?**

=> need to use block cipher in a specific mode

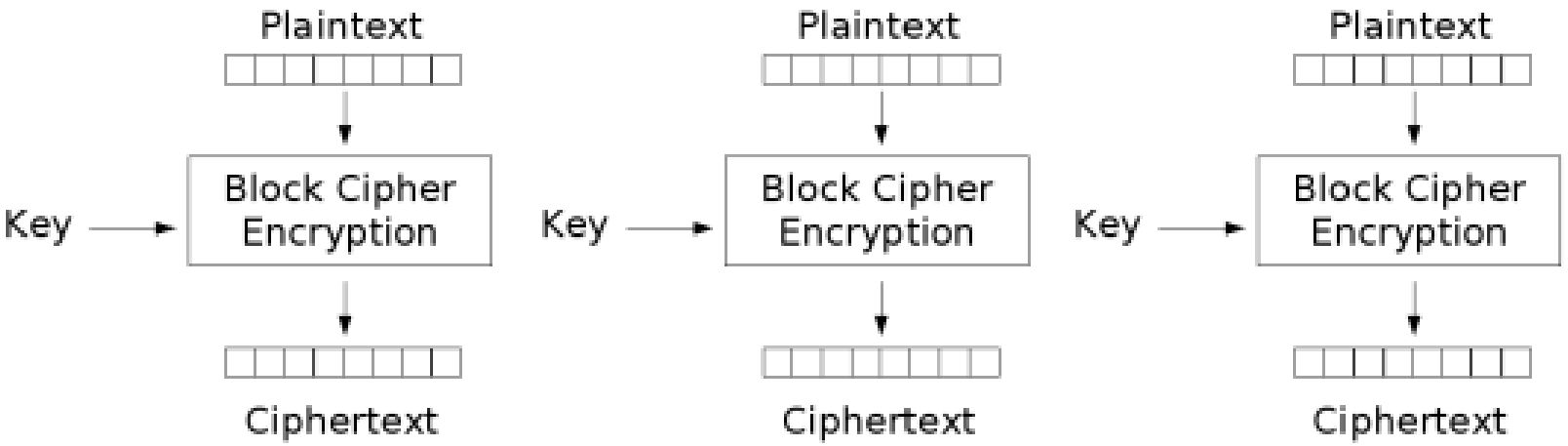
Block Cipher Modes of Operation

- **NIST Special Publication 800-38A (2001)**

  http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

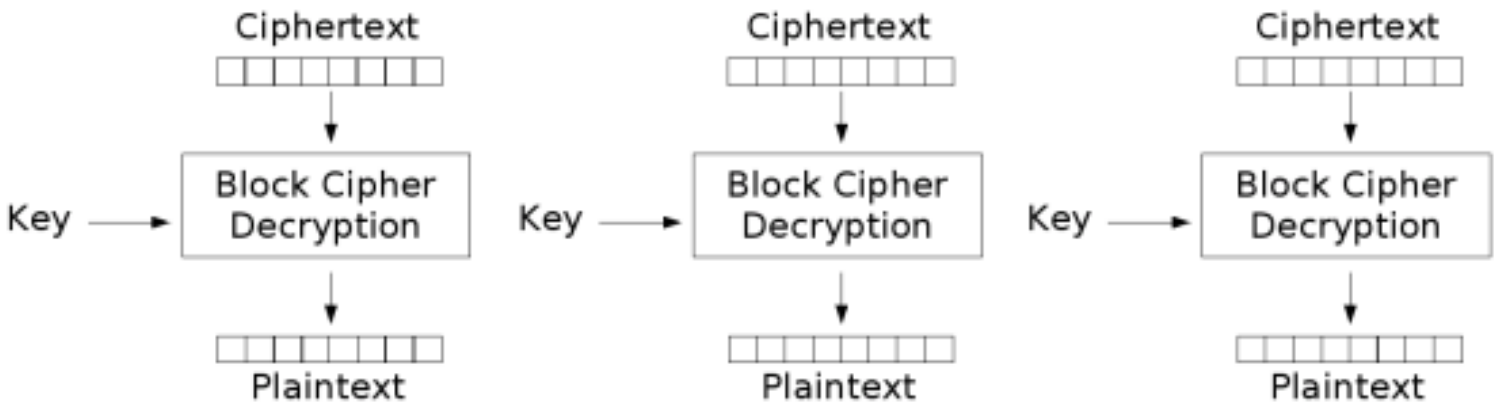
- **Five encryption modes are recommended**
  - **Electronic Codebook (ECB)**
  - **Cipher Block Chaining (CBC)**
  - **Cipher Feedback (CFB)**
  - **Output Feedback (OFB)**
  - **Counter (CTR)**

Electronic Codebook (ECB)



Electronic Codebook (ECB) mode encryption

Electronic Codebook (ECB) Mode



Electronic Codebook (ECB) mode decryption

Electronic Codebook (ECB) Mode

- **Property of ECB mode:**

  the same plaintext block + the same key

  => always the same ciphertext block

  – **If this property is undesirable in an application, ECB mode should not be used**

    - **Example: Data with high redundancy**
      – **Uncompressed image file**
      – **English text in ASCII code. Each character is represented as one byte.**

ASCII code

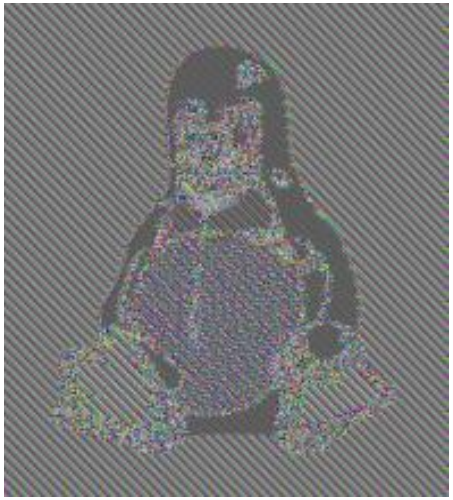
http://en.wikipedia.org/wiki/ASCII

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

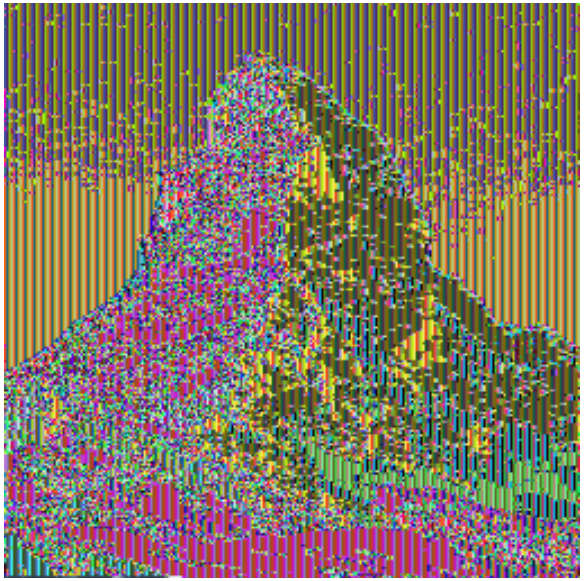
Source: www.LookupTables.com

Electronic Codebook (ECB) Mode

ECB mode encryption

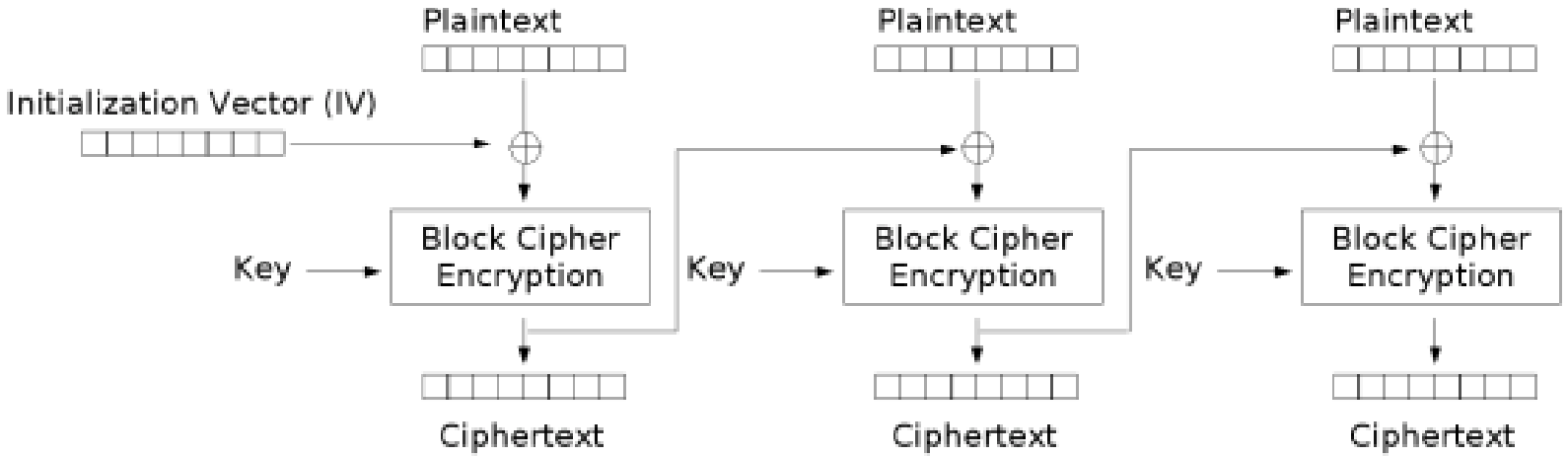


ECB mode encryption



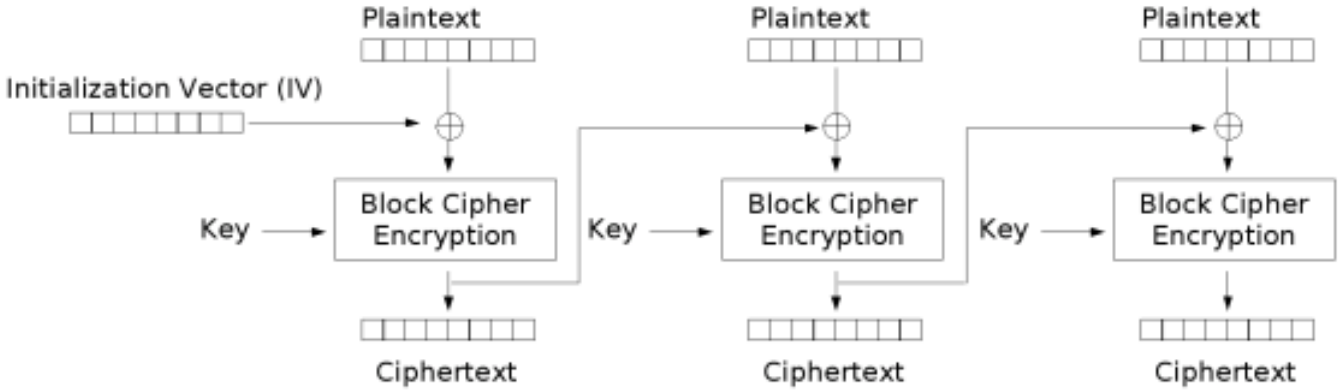
Cipher Block Chaining (CBC) Mode

- **Invented by IBM in 1976**
- **Initialization vector (also called "nonce") is needed**
  - **Nonce need not be secret (normally sent/stored together with ciphertext)**

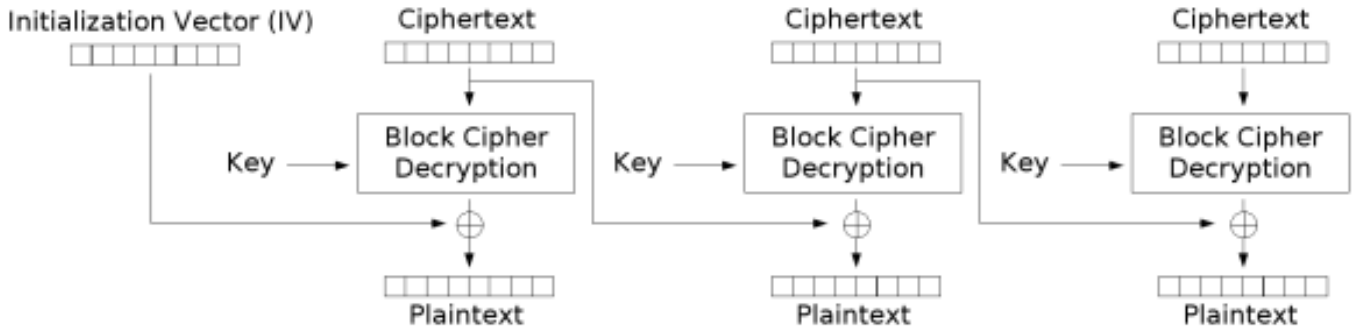


Cipher Block Chaining (CBC) mode encryption

Cipher Block Chaining (CBC) Mode



Cipher Block Chaining (CBC) mode encryption $C_i = E_K(P_i \oplus C_{i-1}), C_0 = IV$



Cipher Block Chaining (CBC) mode decryption $P_i = D_K(C_i) \oplus C_{i-1}, C_0 = IV$

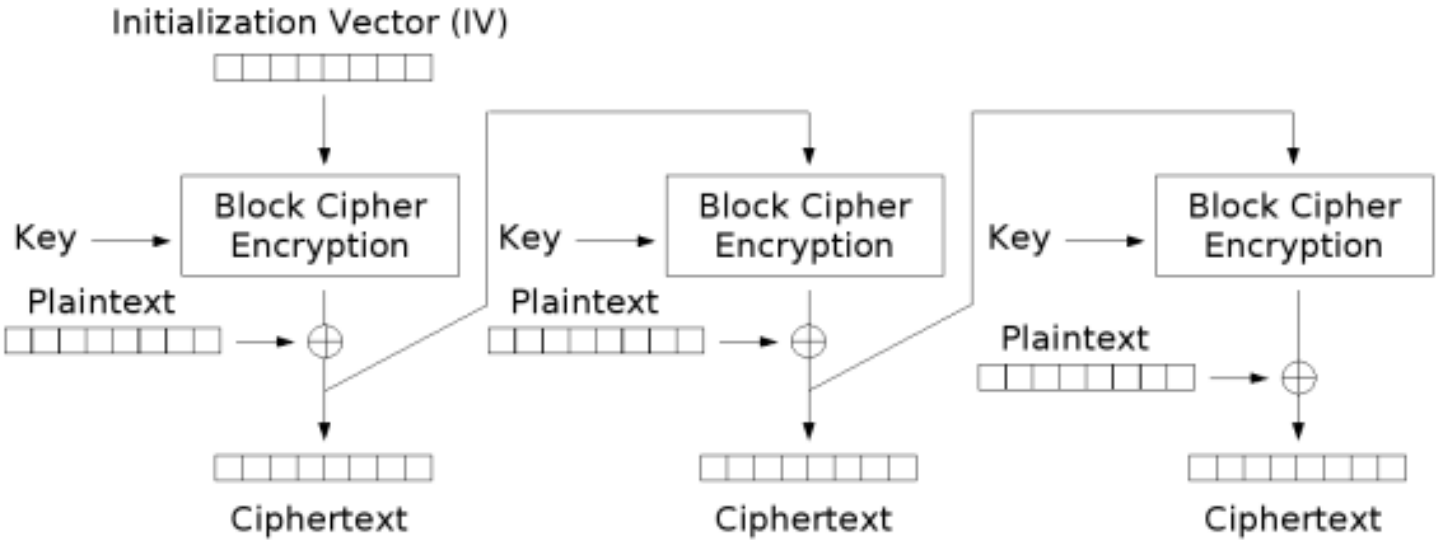
Cipher Block Chaining (CBC) Mode

- **CBC mode, the same key, the same plaintext block at different locations => different ciphertext blocks**
  - **The security of CBC is not that sensitive to the security of IV (If two IVs happen to be the same for the same key, encryption would not fail completely.)**
    - **The most reliable encryption mode!**
    - **A commonly used encryption mode!**

Cipher Feedback (CFB) Mode

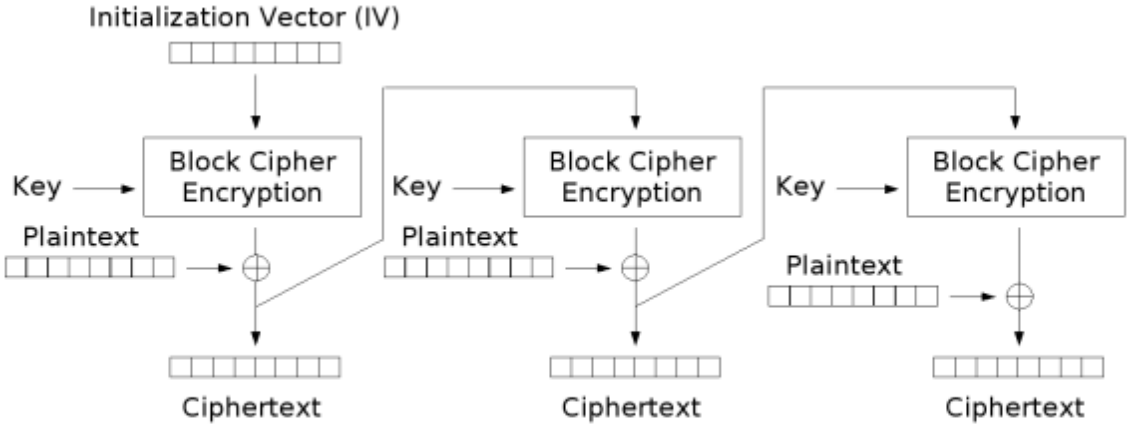
- **A simplified version of CFB:**

$$C_0 = \text{IV}$$
$$C_i = E_K(C_{i-1}) \oplus P_i$$



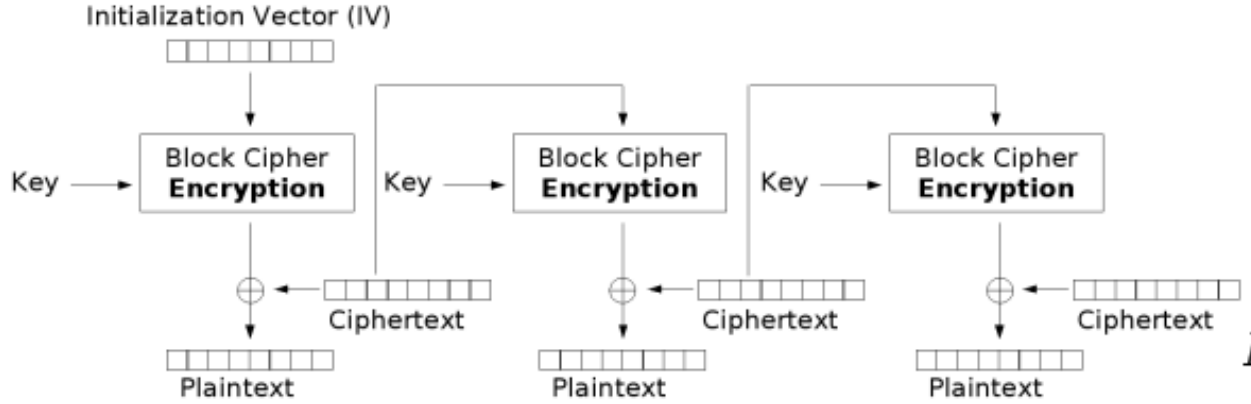
Cipher Feedback (CFB) mode encryption

Cipher Feedback (CFB) Mode



Cipher Feedback (CFB) mode encryption

$$C_0 = \text{IV}$$
$$C_i = E_K(C_{i-1}) \oplus P_i$$

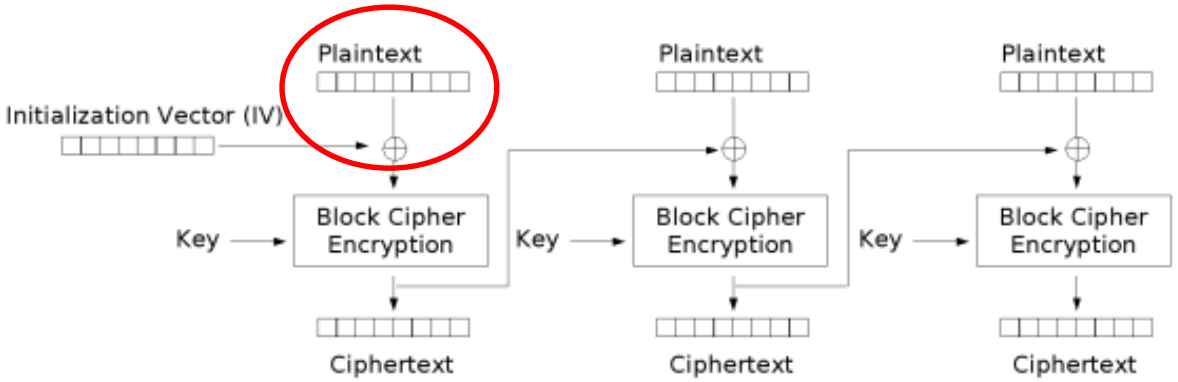


Cipher Feedback (CFB) mode decryption

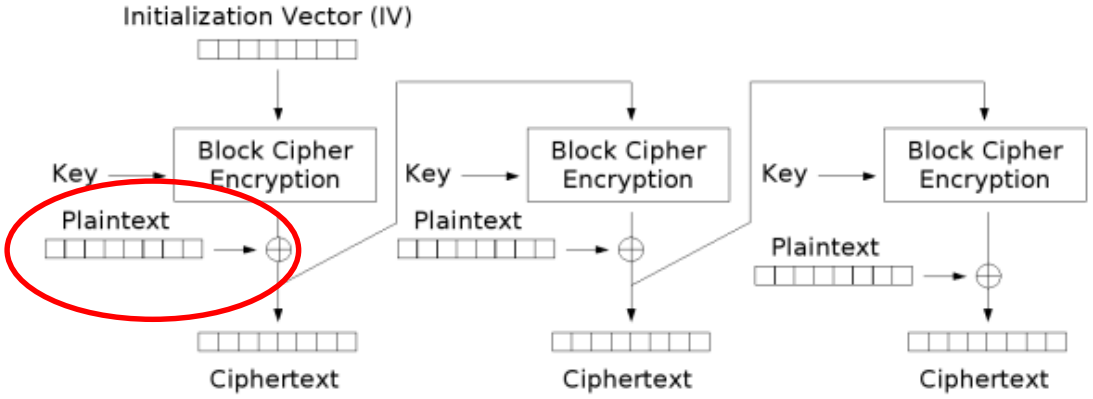
$$C_0 = \text{IV}$$
$$P_i = E_K(C_{i-1}) \oplus C_i$$

Cipher Feedback (CFB) Mode

- Compare CBC & CFB



Cipher Block Chaining (CBC) mode encryption



Cipher Feedback (CFB) mode encryption

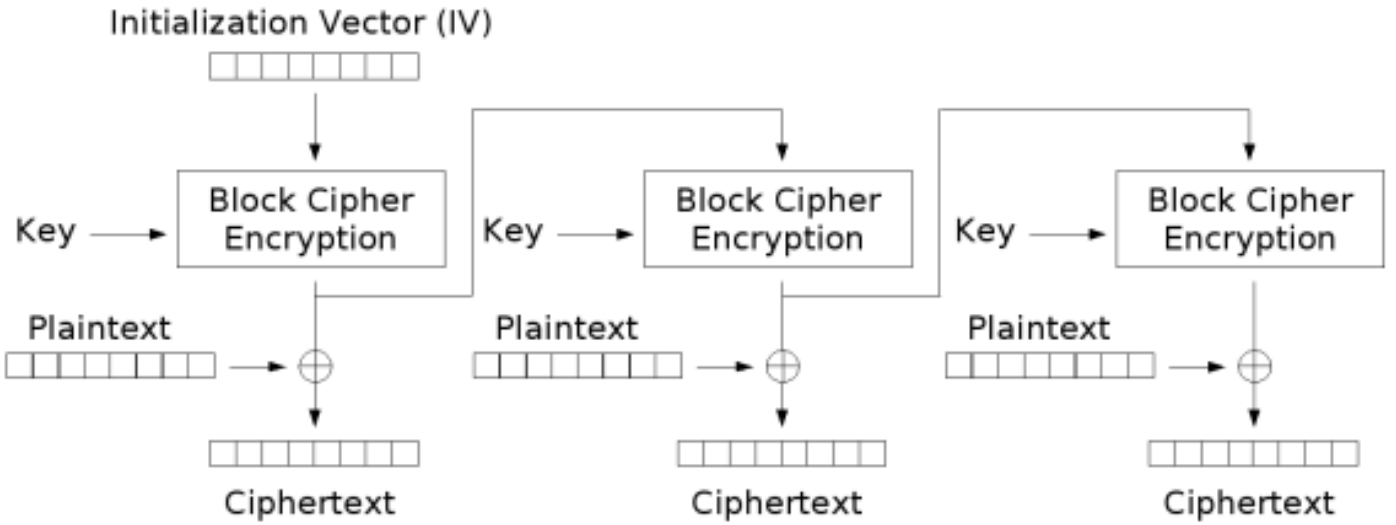
Similarity:
Each ciphertext block is used in the encryption of next block

Difference:
The plaintext in CFB mode is xored with the output of the block cipher

Output Feedback (OFB) Mode

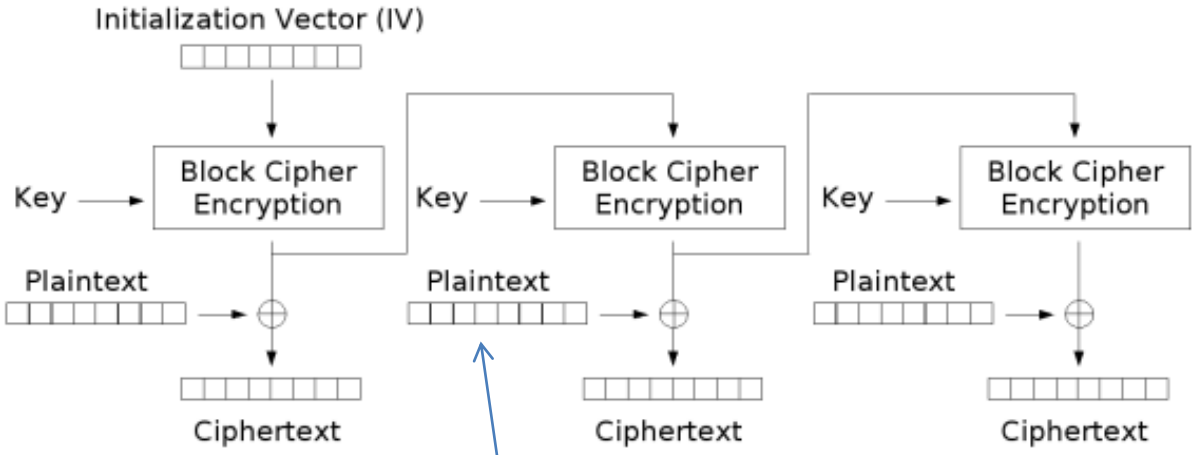
- **A simplified version of OFB:**

$$O_0 = \mathrm{IV}$$
$$O_i = E_K(O_{i-1})$$
$$C_i = P_i \oplus O_i$$

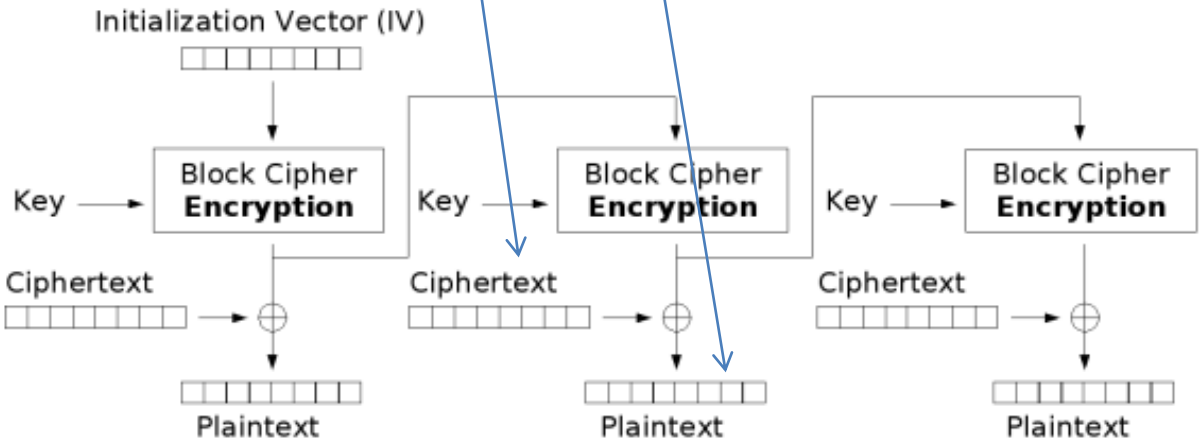


Output Feedback (OFB) mode encryption

Output Feedback (OFB) Mode



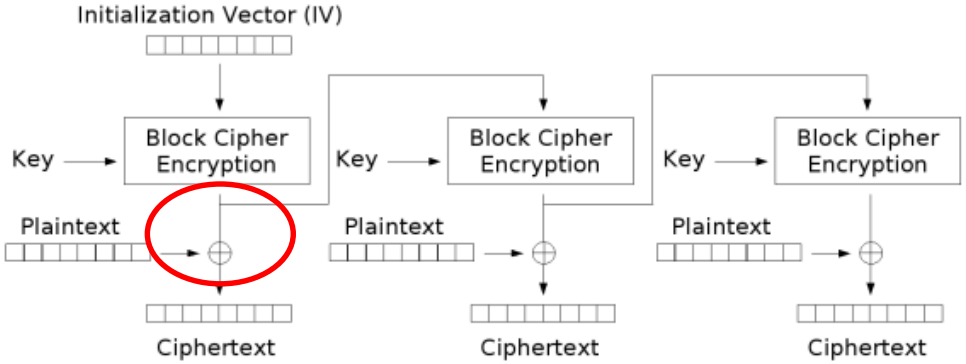
Output Feedback (OFB) mode encryption



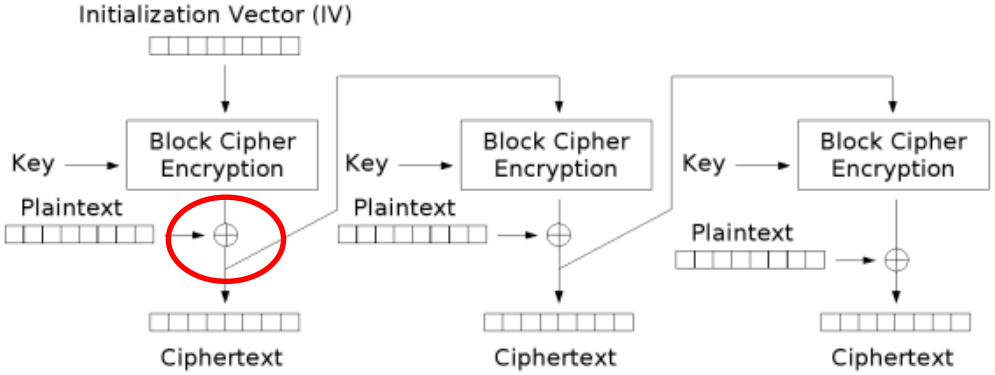
Output Feedback (OFB) mode decryption

Output Feedback (OFB) Mode

- **Compare OFB & CFB**



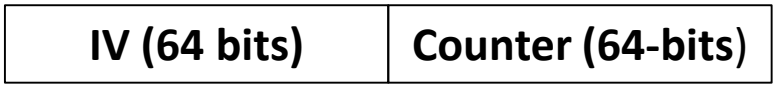
Output Feedback (OFB) mode encryption



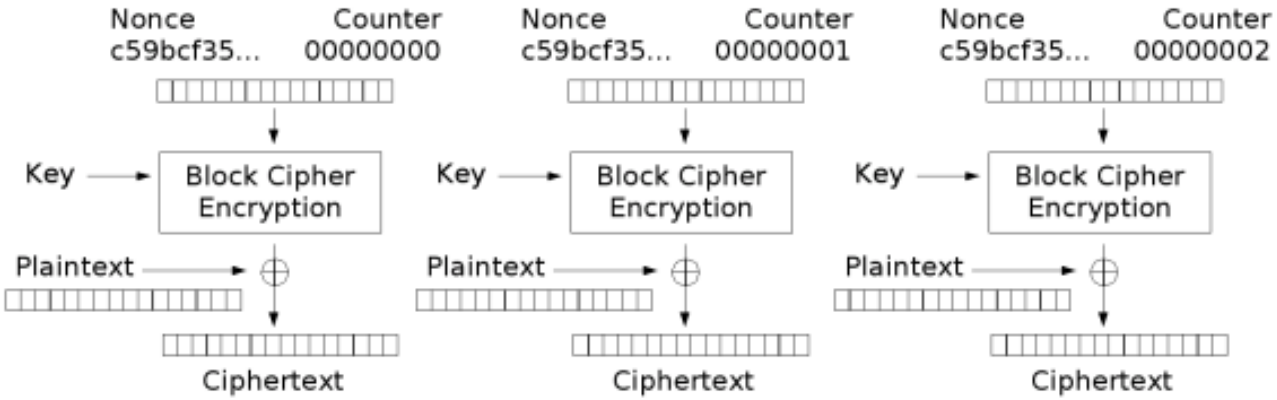
Cipher Feedback (CFB) mode encryption

Counter (CTR) Mode

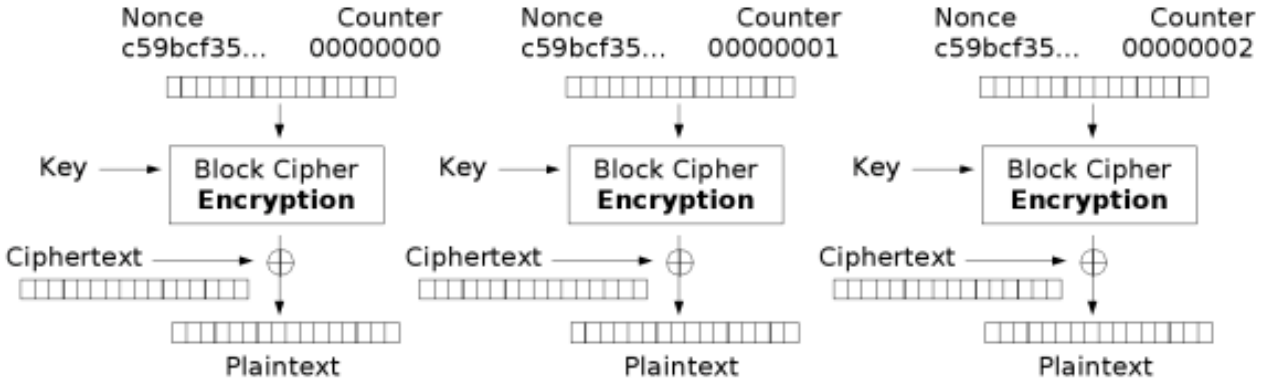
- **For AES-CTR**
  - **The input of block cipher consists of 64-bit IV (nonce) and 64-bit counter**
  - **IV (nonce): different for each message, remains the same for each message**
  - **Counter: its value starts from 0 (for every message), increased by 1 after each block (the counter is computed in the same way for each message)**



Counter (CTR) Mode



Counter (CTR) mode encryption



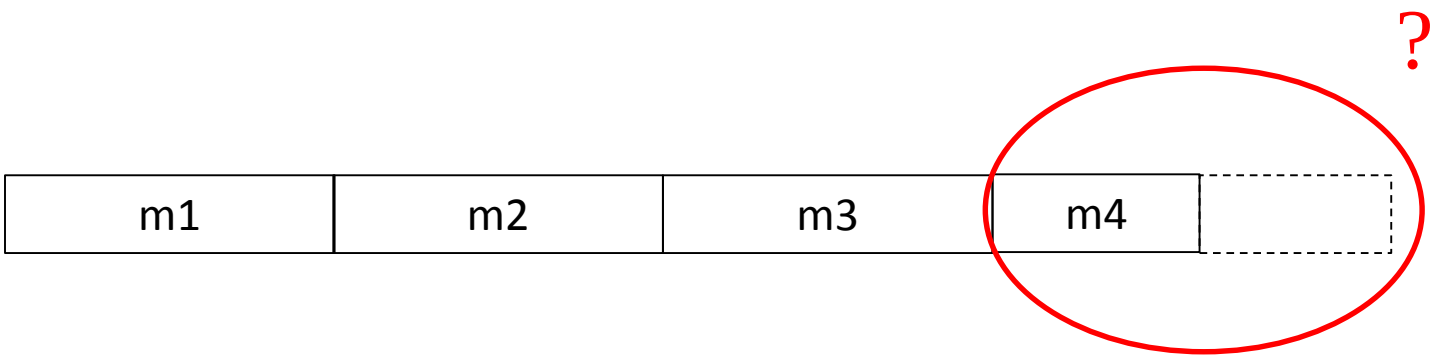
Counter (CTR) mode decryption

Counter (CTR) Mode

- **For each message, the counter should not repeat**
  - **i.e., the length of each message for AES-CTR should not be more than 2^{64} blocks**

How to encrypt a partial block?

- **If the message length is not a multiple of the block size of the block cipher**
  - **the last block is called partial block**



How to encrypt a partial block?

- **ECB & CBC**
  - **Straightforward encryption:**
    - **Pad the partial block to full block**
      - **ECB: padded with random bits**
        - » **Otherwise, the entropy of the partial block may be too small**
      - **CBC: padded with random bits or constant bits**
    - **Ciphertext length larger than plaintext length**
    - **The actual message length is sent together with the ciphertext**

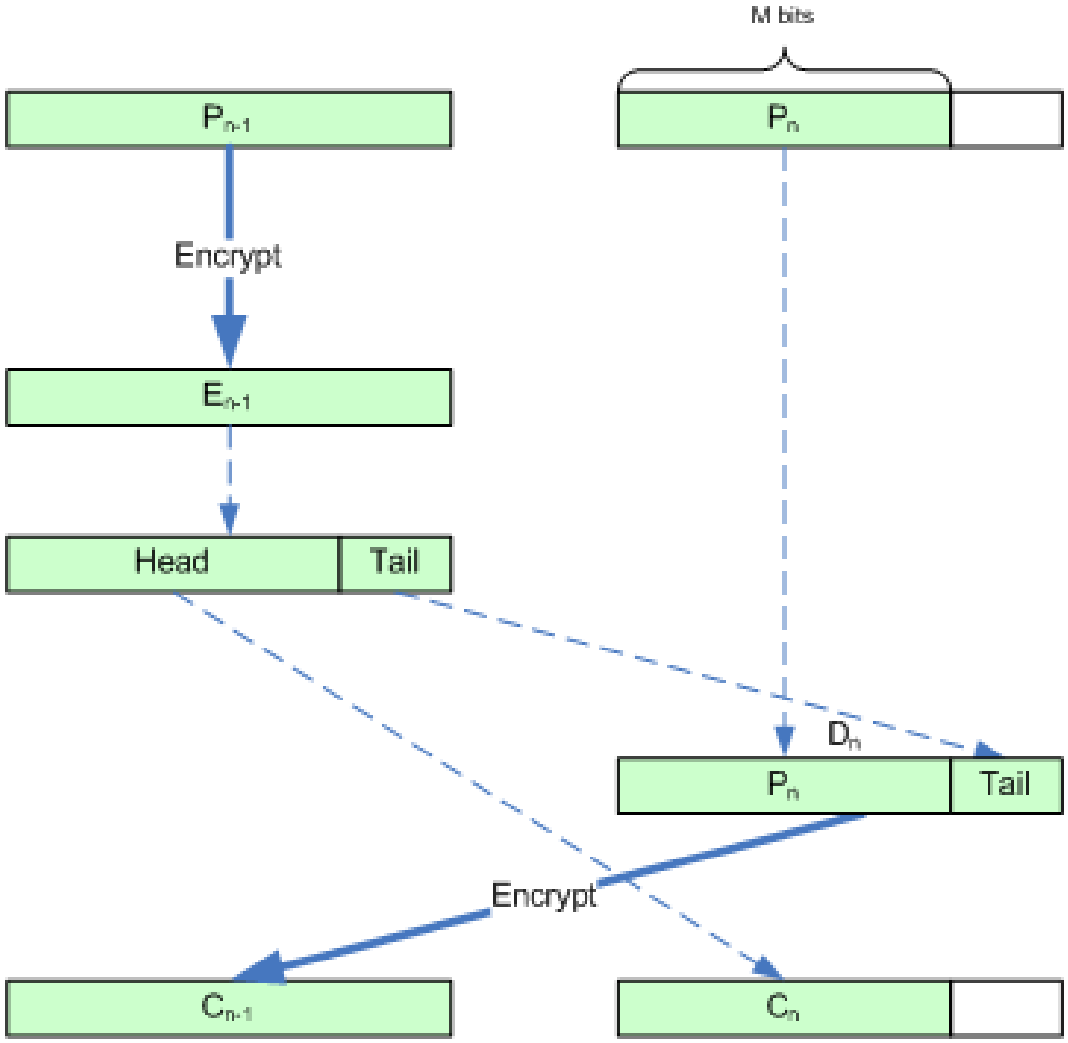
How to encrypt a partial block?

- **CFB, OFB, CTR**
  - **NO partial block problem**
  - **Why?**

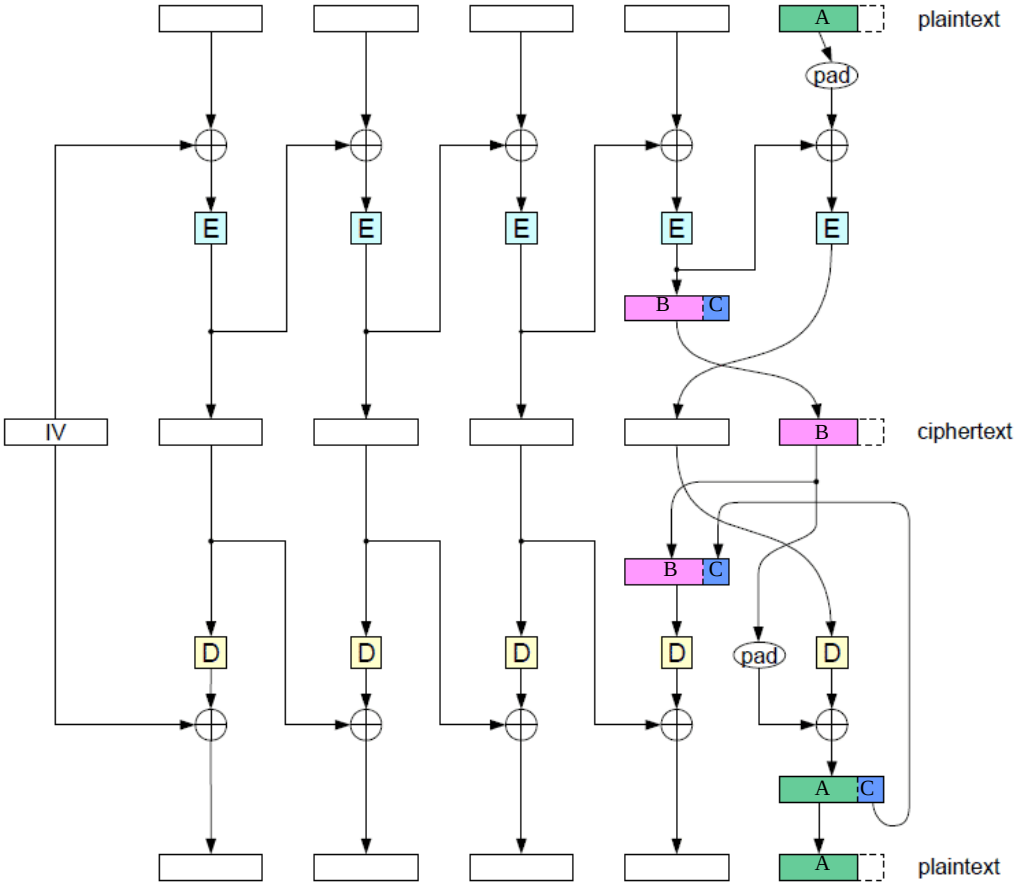
Ciphertext stealing

- **Ciphertext stealing technique**
  - **Try to achieve:**
    - **Ciphertext length = Plaintext length**
  - **ECB ciphertext stealing**
    - **The plaintext should be more than one block**
    - **Otherwise, just use the padding method**
  - **CBC ciphertext stealing**
    - **Not necessary that plaintext is more than one block**
      - **If less than one full block, stealing from C_0 (i.e., from the IV).**

Ciphertext Stealing for ECB mode



Ciphertext Stealing for CBC mode



http://www.scythe.jp/memo/crypto-cts.html

Summary

- **Modes of operations**
  - **ECB: not strong**
    - **Parallel computation is possible**
  - **CBC: strong, the most commonly used**
  - **CFB**
  - **OFB: for the same key, all the IVs must be different**
  - **CTR: for the same key, all the IVs must be different**
    - **Parallel computation is possible**
- **Ciphertext stealing for encrypting partial block**
  - **ECB**
  - **CBC**
  - **Not a problem for CFB, OFB & CTR**