

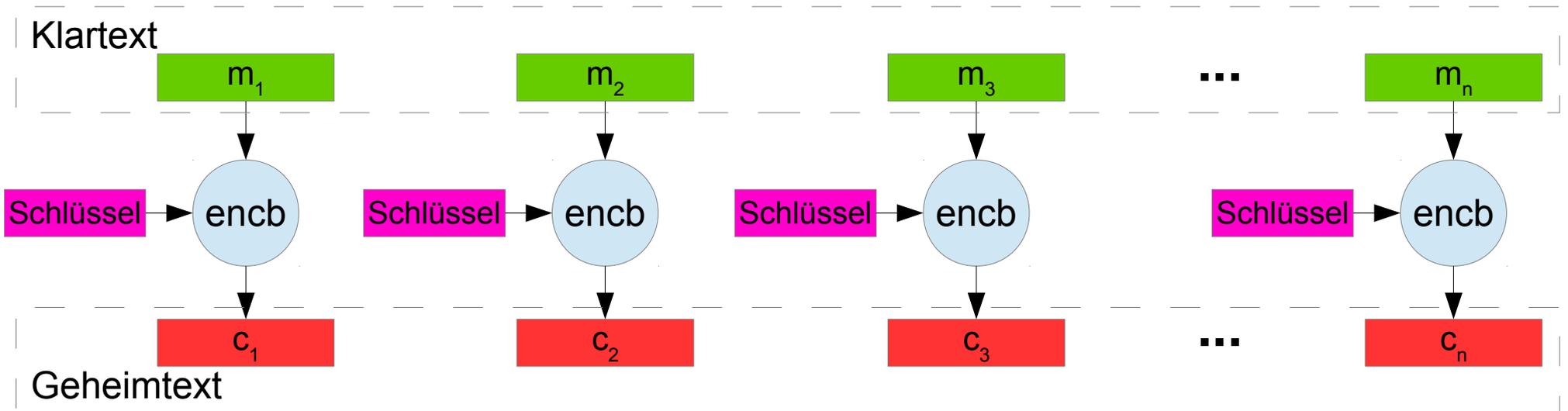
Themen zur Computersicherheit

Symmetrische Blockchiffren

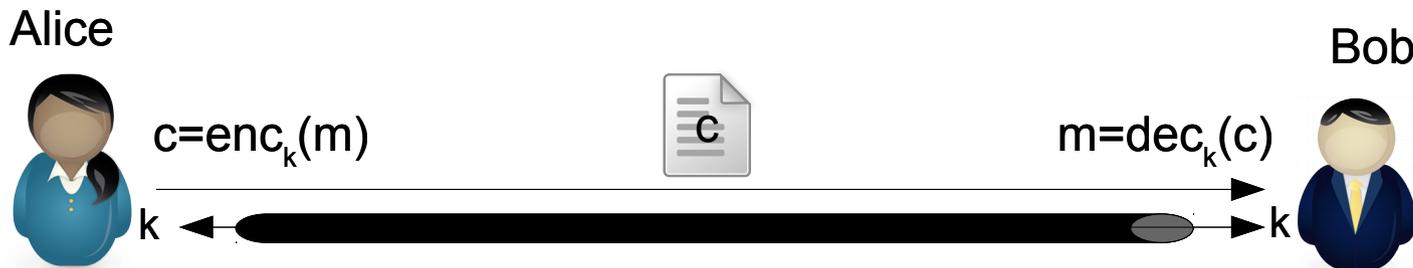
PD Dr. Reinhard Bündgen
buendgen@de.ibm.com

Symmetrische Blockchiffren

- Blockchiffren
 - Verschlüsselung von Nachrichtenblöcken mit Hilfe eines konstanten Schlüssels
 - monoalphabetische Verschlüsselung auf großem Alphabet
 - Klartextlänge muss Vielfaches der Blocklänge sein



- Symmetrische Chiffren
 - $eK=dK (= K)$ und für alle Schlüssel $(ek,dk) \in eK \times dK$ gilt $ek = dk (=k)$



Blockchiffren

- Block: Nachricht der Länge b Bits
- symmetrische Chiffre
 - $\text{enc}_b: K \times \Sigma^b \rightarrow \Sigma^b$, $\text{dec}_b: K \times \Sigma^b \rightarrow \Sigma^b$
 - mit $\text{dec}_b(\text{enc}_b(m)) = m$
- **Ideale Blockchiffre**: verhält sich für jeden Schlüssel wie eine Zufallspermutation, wobei die zu unterschiedlichen Schlüsseln gehörigen Zufallspermutationen unterschiedlich und voneinander unabhängig sind.
- **Angriff auf Blockchiffre**: eine nicht-generische Methode um eine Blockchiffre von einer idealen Blockchiffre zu unterscheiden.
- **Sichere Blockchiffre**: eine Blockchiffre zu der es keinen Angriff gibt.

Designkriterien für Blockchiffren

- Schlüssellänge muss so groß sein, dass eine vollständige Suche praktisch nicht möglich ist.
- **Konfusion:** Die statistischen Eigenschaften des Geheimtextes lassen sich nicht aus den statistischen Eigenschaften des Klartextes ableiten.
- **Diffusion:** Jedes Klartextbit beeinflusst viele Geheimtextbits, jedes Schlüsselbit beeinflusst viele Geheimtextbits
- **Avalanche Kriterium:** Durch die Änderung eines Klartextbit werden rund 50% der Geheimtextbits geändert.
- **Striktes Avalanche Kriterium:** Bei der Änderung eines Klartextbits ändert sich jedes Geheimtextbit mit einer Wahrscheinlichkeit von 50%.

Bekannte Blockchiffren

Chiffre	Autor	Blockgröße	Schlüssellänge	Kommentar
DES	IBM	64	56 + 8 parity	Schlüssel zu kurz
DES2		64	112 + 12 parity	
DES3		64	168 + 24 parity	auch 3DES, DESede
IDEA	Massey & Lai	64	128	
RC5	Rivest	32, 64, 128	128	
Blowfish	Schneier	64	1 – 448	
Camellia	Mitsubishi & NTT	128	128, 192, 256	
AES	Daemen & Rijmen	128	128, 192, 256	NIST Wettbewerb

- DES2 : $k=k_1||k_2$, $\text{des2-enc}_k = \text{des-enc}_{k_1}(\text{des-dec}_{k_2}(\text{des-enc}_{k_1}(m)))$
 - wird mit Sicherheitsniveau von 80 Bits bewertet
- DES3: $k=k_1||k_2||k_3$, $\text{des3-enc}_k = \text{des-enc}_{k_1}(\text{des-dec}_{k_2}(\text{des-enc}_{k_3}(m)))$
 - effektive Schlüssellänge 112 Bits (meet-in-the-middle attack)

Was wenn die Nachricht nicht Blocklänge hat?

- Nachricht kürzer als Blocklänge
 - Padding: Auffüllen auf (Vielfaches von) Blocklänge
 - Schwäche: immer ähnliche Klartextenden
- Nachricht länger als Blocklänge
 - blockweises Chiffrieren
 - Schwäche: monoalphabetische Chiffre über großem Alphabet
- Allgemein: Nachricht kein vielfaches der Blocklänge
 - $m = m_1 || m_2 || \dots || m_n$ mit $|m_i| = b$ für $1 \leq i \leq n-1$, $|m_n| \leq b$
 - Auffüllen des letzten Blocks und blockweises Chiffrieren

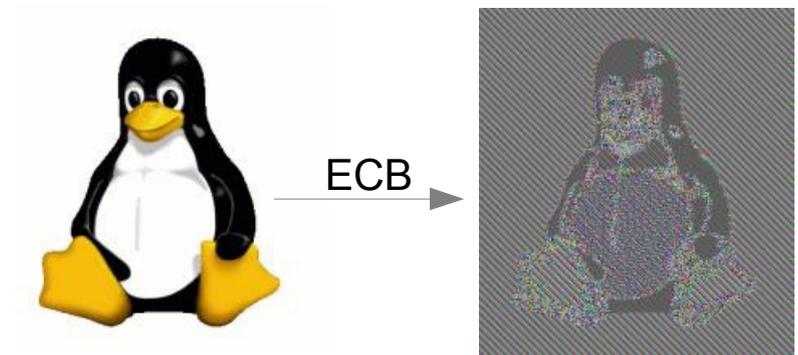
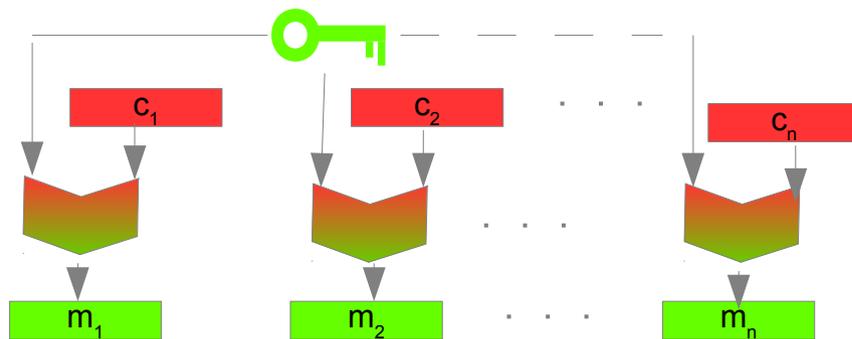
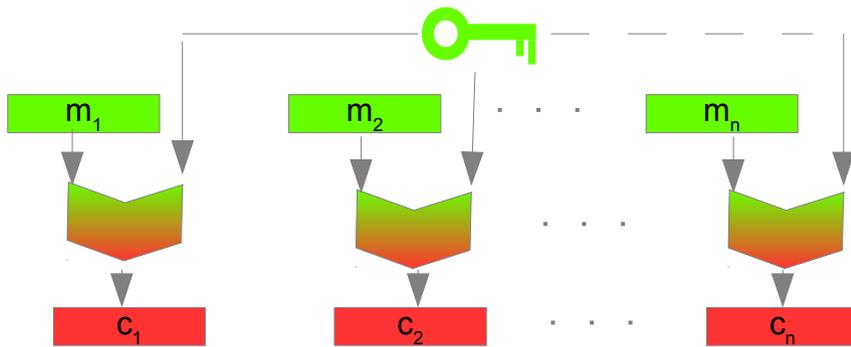
Betriebsmodi

- beschrieben in NIST SP800-38A
 - ECB
 - CBC
 - OFB
 - CFB
 - CTR
- im Folgenden gilt
 - $m = m_1 \parallel m_2 \parallel \dots \parallel m_n$ mit $|m_i| = b$ für $1 \leq i \leq n$
 - $c = c_1 \parallel c_2 \parallel \dots \parallel c_n$ mit $|c_i| = b$ für $1 \leq i \leq n$
 - $|IV| = b$,
 - $|Ctr_i| = b$ für $1 \leq i \leq n$

Betriebsmodus ECB

Electronic Cook Book (ECB) Mode

- Übertragungsfehler bleiben lokal
- wahlfreier Lese- und Schreibzugriff
- anfällig gegen statistische Angriffe

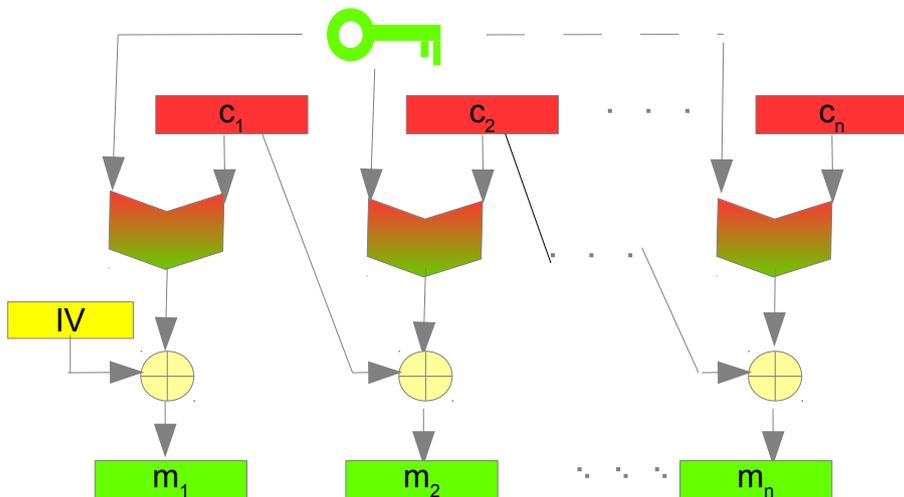
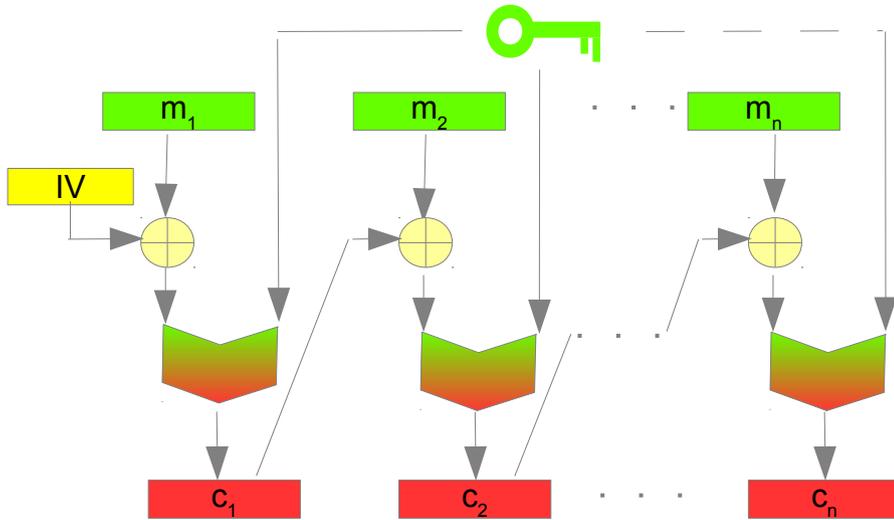


http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation

Betriebsmodus CBC

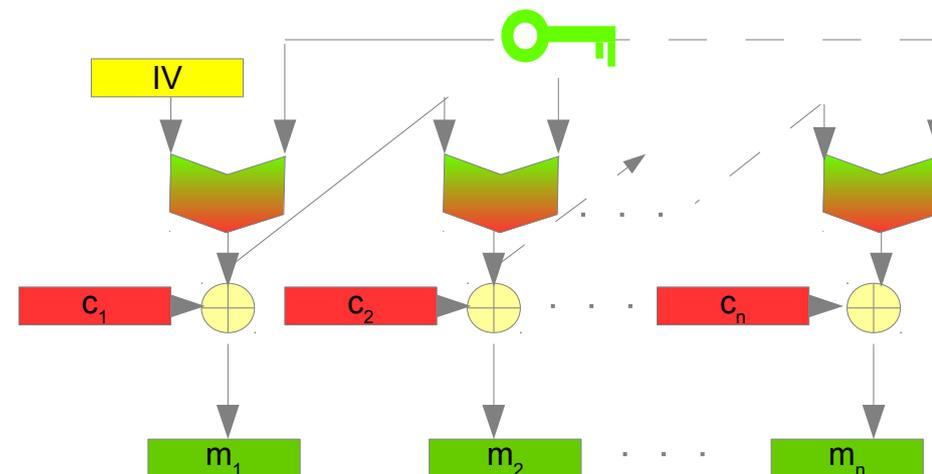
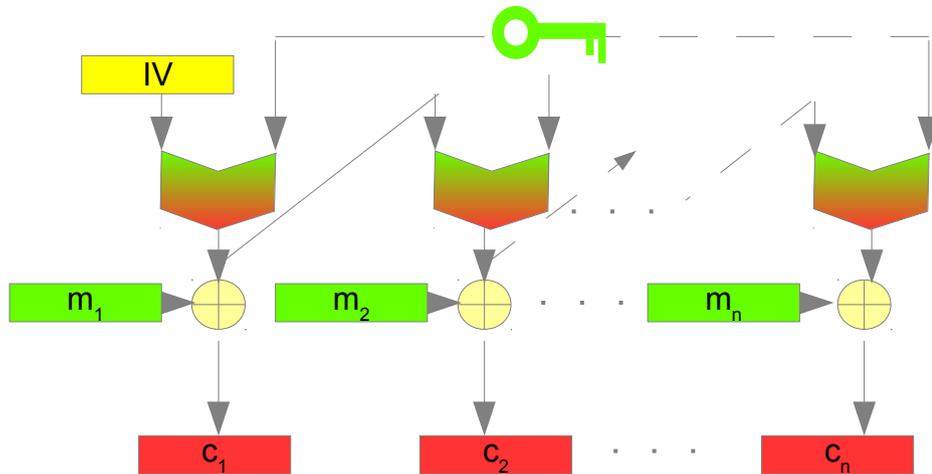
Cipher Block Chaining (CBC) Mode

- benötigt Initialisierungswert (IV)
 - nicht geheim,
 - aber einmalig
- Verschlüsselung
 - $c_0 = IV$
 - $c_i = \text{enc}_k(m_i \oplus c_{i-1})$
- Entschlüsselung
 - $c_0 = IV$
 - $m_i = \text{dec}_k(c_i) \oplus c_{i-1}$
- Übertragungsfehlerverhalten:
 - 1 fehlerhafter Geheimtextblock => 2 fehlerhafte Klartextblöcke
- Zugriffsverhalten
 - fast wahlfreier Lesezugriff
 - Modifikation eines Blockes führt zur Änderung aller folgenden Blöcke
- Sicherheit
 - CPA: so sicher wie zugrundeliegende Blockchiffre
 - CCA: nicht sicher



Betriebsmodus OFB

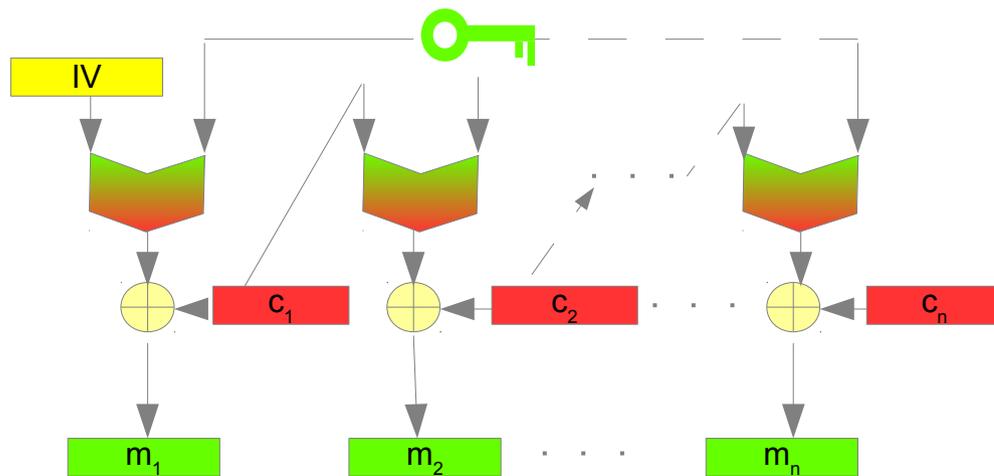
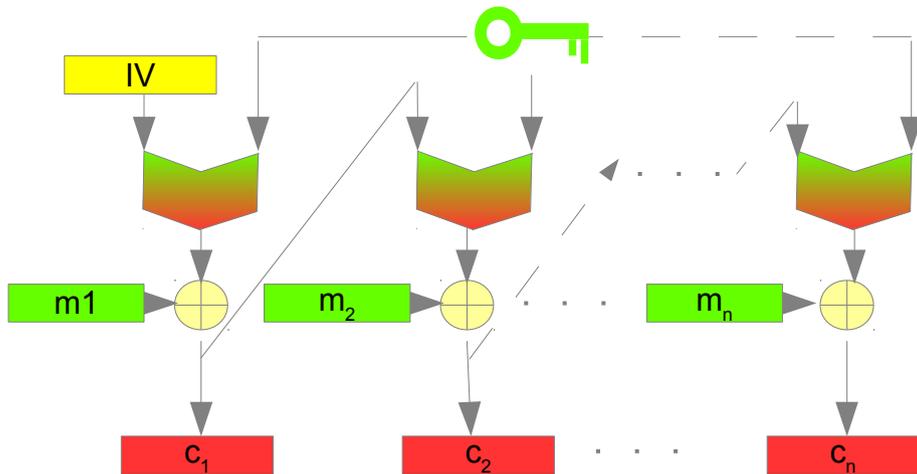
Output Feed Back (OFB) Mode



- benötigt Initialisierungswert (IV)
 - nicht geheim,
 - aber einmalig
- Verschlüsselung
 - $s_0 = IV$
 - $s_i = \text{enc}_k(s_{i-1})$
 - $c_i = s_i \oplus m_i$
- Entschlüsselung
 - $m_i = s_i \oplus c_i$
- s_0, s_1, \dots ist Pseudozufallsfolge
- stromorientierter Modus
- Übertragungsfehlerverhalten:
 - 1 fehlender Geheimtextblock => folgende Blöcke werden falsch entschlüsselt
- Sicherheit
 - CPA: so sicher wie zugrundeliegende Blockchiffre
 - CCA: nicht sicher

Betriebsmodus CFB

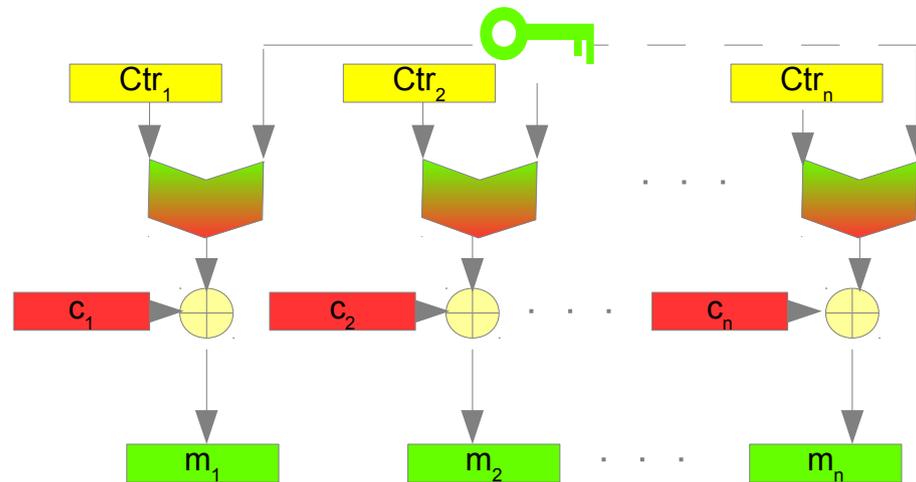
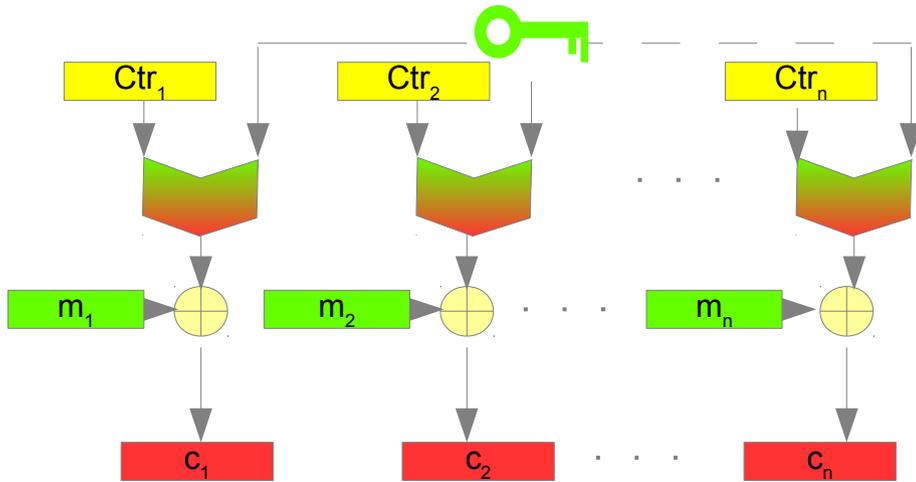
Cipher-Feedback-Modus (CFB_{blksize})



- benötigt Initialisierungswert (IV)
 - nicht geheim,
 - aber einmalig
- Verschlüsselung
 - $c_0 = IV$
 - $c_i = \text{encb}_k(c_{i-1}) \oplus m_i$
- Entschlüsselung
 - $c_0 = IV$
 - $m_i = \text{encb}_k(c_{i-1}) \oplus c_i$
- stromorientierter Modus
- Fehlerverhalten:
 - 1 fehlerhafter Geheimtextblock => 2 fehlerhafte Klartextblöcke
- Sicherheit
 - CPA: so sicher wie zugrundeliegende Blockchiffre
 - CCA: nicht sicher

Betriebsmodus CTR

Counter-Betriebsmodus (CTR)



- benötigt Zähler Werte (Ctr_1, \dots, Ctr_n)
 - nicht geheim,
 - aber einmalig
 - Ctr_1 gegeben
 - $Ctr_i = \text{inc}(Ctr_{i-1})$
- Verschlüsselung
 - $c_i = \text{enc}_k(ctr_i) \oplus m_i$
- Entschlüsselung
 - $m_i = \text{enc}_k(ctr_i) \oplus c_i$
- $\text{enc}_k(ctr_1), \text{enc}_k(ctr_2), \dots$ ist Pseudozufallsfolge
- stromorientierter Modus
- erlaubt wahlfreien Zugriff auf jeden Block
- parallelisierbar
- Fehlerverhalten:
 - 1 fehlender Geheimtextblock => folgende Blöcke werden falsch entschlüsselt
- $Ctr_i = \text{nonce} || ctr_i$ mit $|ctr_i| = s, |\text{nonce}| = b-s$
 - $\text{inc}(Ctr_i) = \text{nonce} || ((1+ctr_i) \bmod 2^s)$
- Sicherheit: CPA: so sicher wie zugrundeliegende Blockchiffre

Padding

Klartext: $m = m_1 \parallel m_2 \parallel \dots \parallel m_n$ mit $|m_i| = b$ für $1 \leq i \leq n-1$, $|m_n| \leq b$

Naives Padding:

- $|m_n| = j < b \Rightarrow m'_n = m_n \parallel 0^{b-j}$
- Nachteile:
 - Länge des Klartextes kann dem Geheimtext nicht entnommen werden
 - die meisten Klartexte mit Padding Zeichen: möglicher Angriffspunkt

Byte-Padding gemäß PKCS#5/RFC 1423 bzw PKCS#7/RFC 2315

- B Blockgröße in Bytes, sei $|m_n| = J$ Bytes
- falls $J < B$ Bytes dann $m'_n = m_n \parallel \text{byte}(B-J)^{B-J}$ ersetzt m_n
- falls $J = B$ Bytes dann hänge $m_{n+1} = \text{byte}(B)^B$ an Nachricht an
- Nachteil
 - Geheimtext kann ein Block länger sein als Klartext

Notation

- X^k : k-fache Wiederholung von X
- $\text{byte}(k)$: Darstellung der Zahl k als Byte (ohne Vorzeichen)

Beispiel

- Blockgröße 16, Hex-String S1 = „a125“
 - naives Padding von S1: „a1250000 00000000 00000000 00000000“
 - naives Padding von msb(30,S1): „a1240000 00000000 00000000 00000000“
 - PKCS #5/7 Padding von S1: „a1240c0c 0c0c0c0c 0c0c0c0c 0c0c0c0c“
- Blockgröße 8 Hex-String S2 = „01020304 05060708“
 - naives Padding von S2: „01020304 05060708“
 - PKCS #5/7 Padding von S2: „01020304 05060708 08080808 08080808“

Cipher Stealing (CBC-CS)

- Beschrieben in NIST SP 800-38A addendum

- Klartext: $m = m_1 \parallel m_2 \parallel \dots \parallel m_n$ mit $|m_i| = b$
für $1 \leq i \leq n-1$, $|m_n| = j \leq b$

- $m' = m \parallel 0^{b-j}$

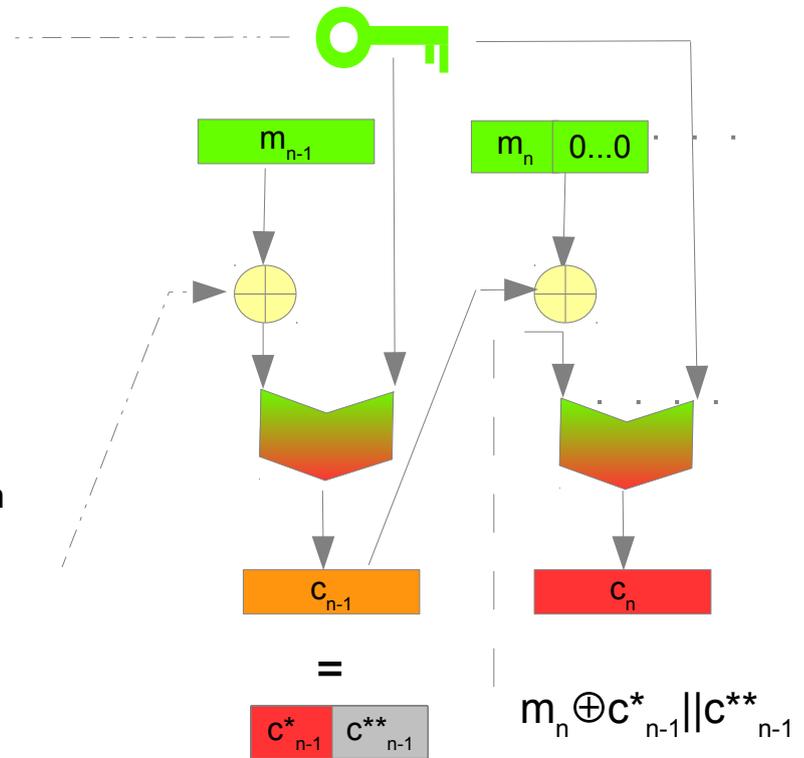
- $c' = \text{enc-cbc}_k(m') = c_1 \parallel c_2 \parallel \dots \parallel c_{n-1} \parallel c_n$

- $c = \text{enc-cbc-cs}_k(m) = c_1 \parallel c_2 \parallel \dots \parallel c_{n-1}^* \parallel c_n$
mit $c_{n-1}^* = \text{msb}(j, c_{n-1})$

- die Information über $\text{lsb}(b-j, c_{n-1})$
geht nicht verloren:

- $\text{dec}_k(c_n) = m_n \oplus \text{msb}(j, c_{n-1}) \parallel 0^{b-j} \oplus \text{lsb}(b-j, c_{n-1})$

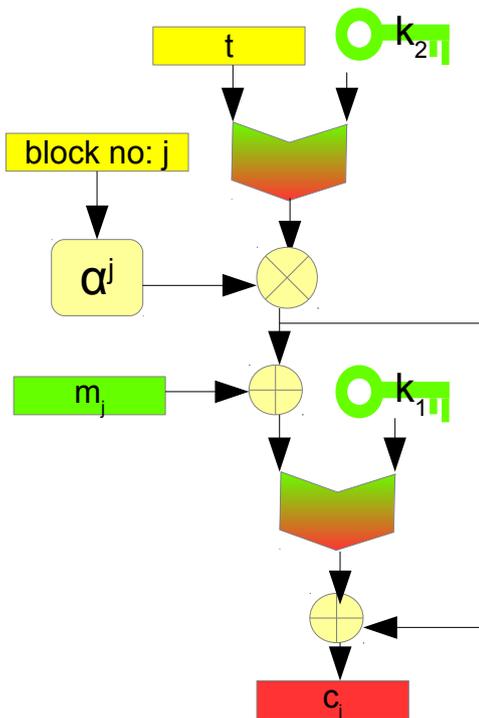
- $c_{n-1} = \text{msb}(j, c_{n-1}) \parallel \text{lsb}(b-j, c_{n-1}) = c_{n-1}^* \parallel \text{lsb}(b-j, \text{dec}_k(c_n))$



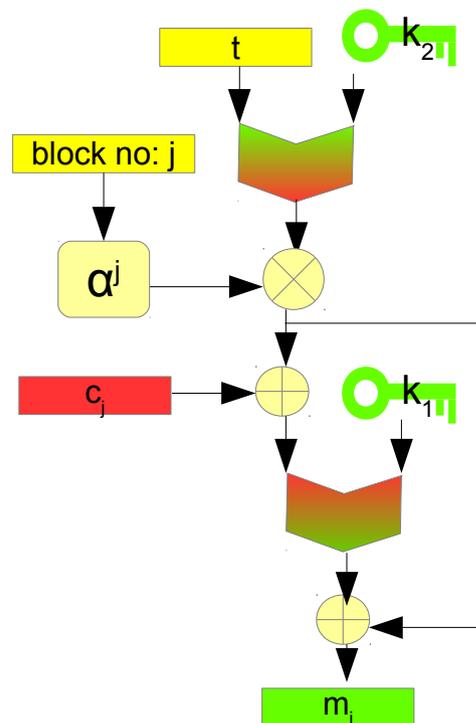
XTS-AES eine Blockchiffre für gespeicherte (ruhende) Daten

- Xor-encrypt-xor (XEX)-based Tweakable block cipher with cipher text Stealing
- NIST SP800-38E & IEEE P1619-2007
- Parameter: Chiffre (AES-128, AES-256), 2 AES Schlüssel (k_1, k_2), ein Tweak $t \in \{0,1\}^{128}$
- Einschränkung: mit gleichem Schlüssel und Tweak gespeicherte Dateneinheit: $16 - 2^{24}$ Bytes
- Falls Nachricht nicht Vielfaches der Blocklänge: cipher stealing für letzte 2 (angebrochene) Blöcke
- wahlfreier Zugriff auf jeden Block

XTS Blockverschlüsselung von Block j



XTS Blockentschlüsselung von Block j



Notation

α^j

Operation in $GF(2^{128})$
mit α primitives Element
in $GF(2^{128})$

\otimes

Multiplikation in
 $GF(2^{128})$