



COMPUTERSYSTEMSICHERHEIT

Thema 3: Symmetrische Kryptographie

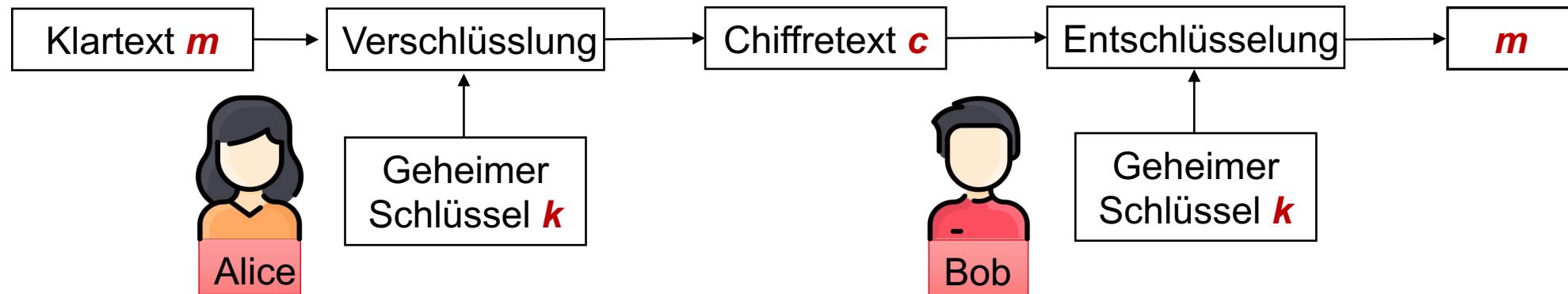
Prof. Sebastian Faust

THEMENÜBERSICHT

- Motivation Symmetrische Kryptographie
- Definition symmetrische Chiffre
- One-Time-Pad
- Blockchiffren
- Modes of Operation
- Kryptographische Hashfunktionen
- Message Authentication Codes (MACs)
- Authenticated Encryption

SYMMETRISCHE KRYPTOGRAPHIE

- **Symmetrisch Kryptographie:** Es gibt nur **einen** Schlüssel für alle Algorithmen



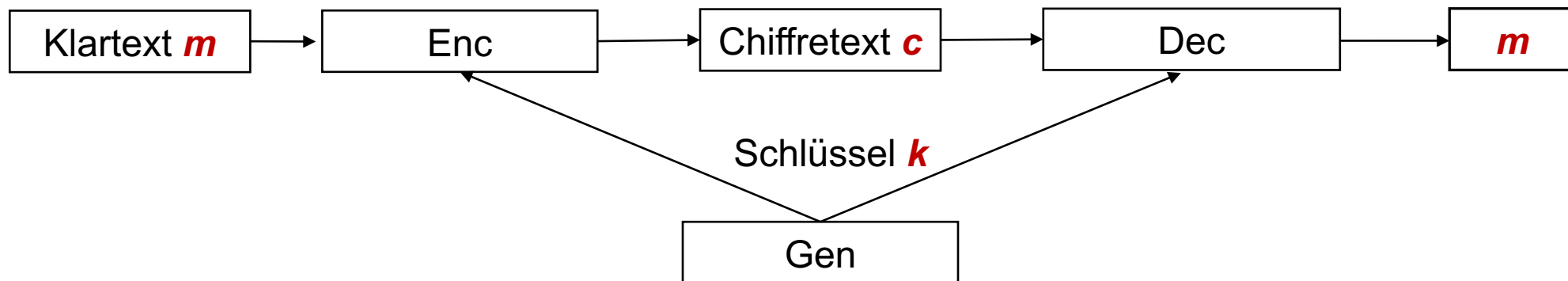
- Einigung auf geheimen Schlüssel:
 - Physikalisches Treffen
 - Kryptographisches Protokoll (später mehr)



DEFINITION SYMMETRISCHER CHIFFREN

FUNKTIONALE DEFINITION

- **Funktionale Definition:** Beschreibt Input/Output-Verhalten der Algorithmen
- **Algorithmen:** (Gen, Enc, Dec)

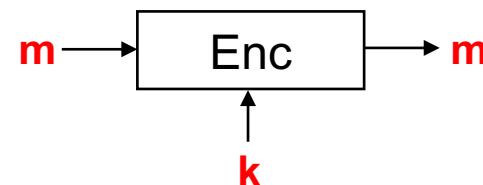


- **Korrektheit:** Die Entschlüsselung eines gültigen Chiffretexts resultiert in die original verschlüsselte Nachricht
 $\rightarrow Dec(k, Enc(k, m)) = m$ für alle Nachrichten *m* und Schlüssel $k \leftarrow Gen$
- **Effizienz:** Verschlüsselung und Entschlüsselung sind effizient ($> 1\text{GB/s}$)

SICHERHEITSDEFINITION

- **Ziel des Angreifers:** Was ist ein erfolgreicher Angriff?

- *Naive Option:* Angreifer lernt den Schlüssel **k** nicht



- *In der Kryptographie:* Angreifer lernt „**nichts Neues**“ über **m**



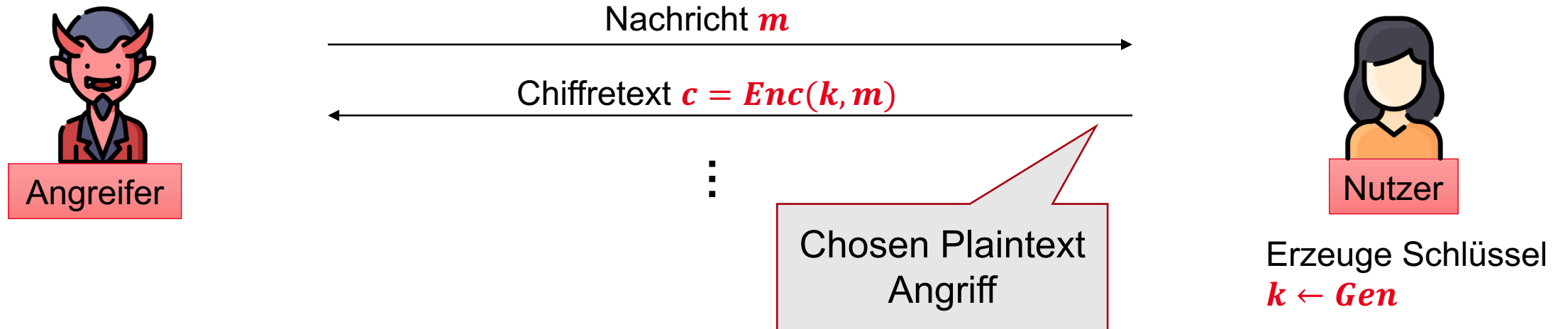
- **Angreifermodell:** Was kann der Angreifer tun und sehen?

- Angreifer lernt nur Chiffretexte (*known ciphertext attack*), z.B. durch Abhören des Kanals
 - Angreifer lernt Paare von Klartexten/Chiffretexten (*known plaintext/ciphertext attack*), z.B. bestimmter Teil der verschlüsselten Nachricht kann bekannt sein
 - Angreifer wählt Klartexte und lernt zugehörige Chiffretexte (*chosen plaintext attack*), z.B. Angreifer Nutzer davon überzeugen Nachrichten seiner Wahl zu verschlüsseln



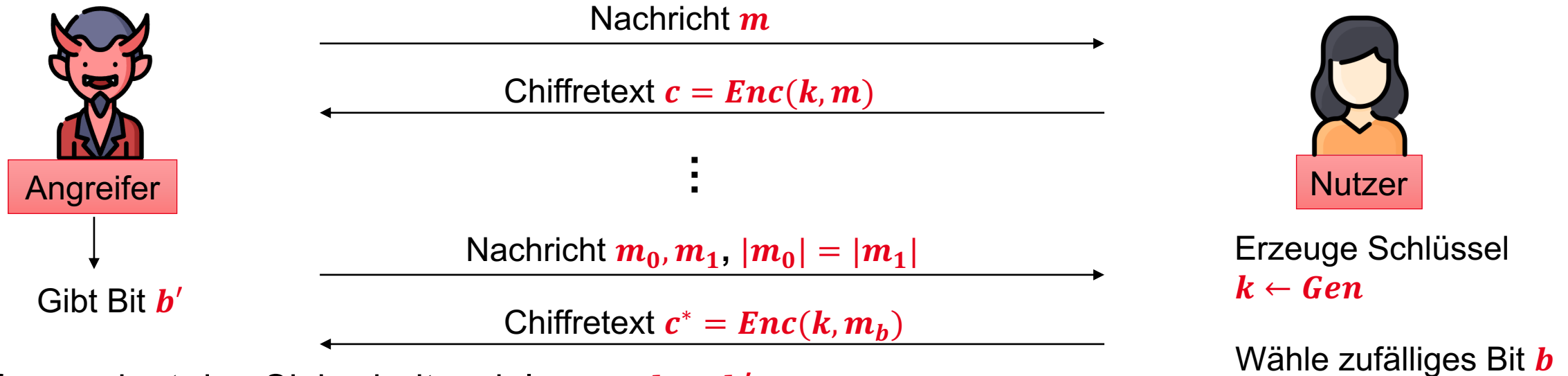
SICHERHEITSSPIEL (IND-CPA)

Sicherheit wird in der Kryptographie durch ein **Spiel** zwischen **Angreifer** und „**Nutzer**“ definiert



SICHERHEITSSPIEL (IND-CPA)

Sicherheit wird in der Kryptographie durch ein **Spiel** zwischen **Angreifer** und „**Nutzer**“ definiert



Angreifer gewinnt das Sicherheitsspiel wenn $b = b'$

Sicherheit: Symmetrische Chiffre ist **IND-CPA sicher**, falls alle „effizienten“ Angreifer das Sicherheitsspiel maximal mit Wahrscheinlichkeit $\approx \frac{1}{2}$ gewinnen können

IMPLIKATIONEN

Falls Verschlüsselungsverfahren IND-CPA sicher, dann:

- Das Beste was der Angreifer tun kann ist b raten
- Angreifer kann Chiffretext c^* im wesentlichen ignorieren
- Chiffretext c^* gibt keine zusätzlichen Informationen über verschlüsselte Nachricht

WIE ZEIGEN WIR (UN)SICHERHEIT?

Welche Annahmen machen wir über die Strategie des Angreifers? **Keine**, außer dass sie effizient sind!

Was bedeutet „**effizient**“?

- **Nicht effizient:** hat Laufzeit exponentiell in der Schlüssellänge, bricht Faktorisierung,...
- **Effizient:** siehe „Einführung in die Kryptographie“

Unsicheres Verfahren: Konstruiere effizienten Angreifer, der mit Wahrscheinlichkeit $> \frac{1}{2}$ das Sicherheitsspiel gewinnt

Sicheres Verfahren: Zeige, dass alle effizienten Angreifer das Sicherheitsspiel mit Wahrscheinlichkeit $< \frac{1}{2}$ gewinnen

BRUTE FORCE ANGRIFF

Betrachte einen Brute Force Angriff wo der Angreifer alle Schlüssel ausprobiert, um das Verschlüsselungsverfahren anzugreifen. Bricht dieser Angreifer die IND-CPA Sicherheitseigenschaft?

- Ja, der Angreifer kann m_0 von m_1 unterscheiden.
- Nein, der Angreifer lernt nichts von c^* über m_b .
- Nein, der Angreifer ist nicht effizient.



pingo.coactum.de
Nr.: 643569

ONE-TIME PAD VERSCHLÜSSELUNG

WIEDERHOLUNG: XOR

- Bit XOR Operationen

$x \oplus 0 = x$
$x \oplus x = 0$
$x \oplus y = y \oplus x$
$(x \oplus y) \oplus z = x \oplus (y \oplus z)$
$(x \oplus y) \oplus x = y$

Erweiterung auf Bitstrings

1	0	0	1	0	1	1	1
\oplus							
1	1	1	0	1	0	1	0
\oplus							
0	1	1	1	1	1	0	1

ONE-TIME PAD

Wird häufig auch Vernam Chiffre genannt nach Gilbert Vernam
One-Time Pad zur Verschlüsselung von Bitstrings der Länge n

- **Gen:** Ausgabe zufälliger Schlüssel $k \xleftarrow{R} \{0, 1\}^n$
- **Enc:** Für $m \in M$: Ausgabe $\text{Enc}(k, m) = k \oplus m$.
- **Dec:** Für $c \in C$: Ausgabe $\text{Dec}(k, c) = k \oplus c$.



Gilbert Vernam
(1890-1960)

Beispiel:

$$\begin{array}{rcl} m = & \boxed{1} & \boxed{0} & \boxed{0} & \boxed{1} & \boxed{0} & \boxed{1} & \boxed{1} & \boxed{1} \\ & & & & \oplus & & & & \\ k = & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{0} & \boxed{1} & \boxed{0} & \boxed{1} & \boxed{0} \\ & & & & = & & & & \\ c = & \boxed{0} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{0} & \boxed{1} \end{array}$$

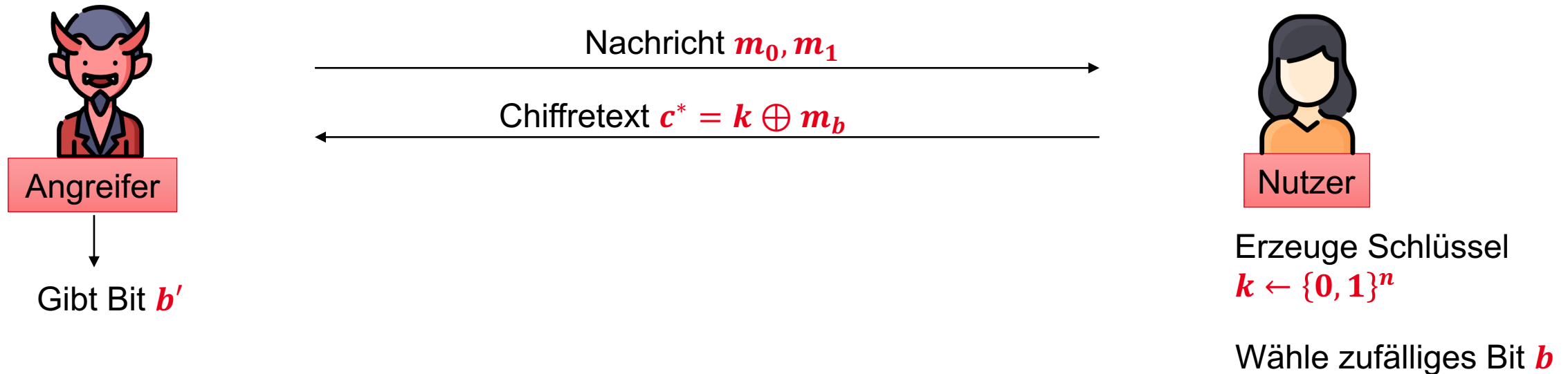
Korrektheit

Für jedes k und m gilt:

$$\begin{aligned} \text{Dec}(k, \text{Enc}(k, m)) &= \\ \text{Dec}(k, m \oplus k) &= \\ k \oplus m \oplus k &= m. \end{aligned}$$

SICHERHEIT: ONE-TIME PAD

Beschränktes Sicherheitsspiel: Angreifer erhält keinen Zugriff auf Chiffretexte (wichtig: siehe nächste Slide)



Angreifer gewinnt das Sicherheitsspiel wenn $b = b'$

Perfekte Sicherheit: c^* gibt keine Information über m_b preis



Dieser Artikel erläutert die Kommunikationsverbindung, für das Geschicklichkeitsspiel siehe [Heißer Draht \(Spiel\)](#), für die [Samstagabendshow](#) siehe [Der heiße Draht](#), für die Zeitung siehe [Der heisse Draht](#).

Als **Heißer Draht** ([engl. hotline](#)) oder **Rotes Telefon** wurde eine ständige [Fernschreiberverbindung](#) zwischen der [Sowjetunion](#) und den [Vereinigten Staaten](#) während der Zeit des [Kalten Krieges](#) bezeichnet.

Inhaltsverzeichnis [\[Verbergen\]](#)

- 1 [Einrichtung](#)
- 2 [Erster Einsatz](#)
- 3 [Reaktivierung](#)
- 4 [Sonstiges](#)
- 5 [Literatur](#)
- 6 [Einzelnachweise](#)

Einrichtung [\[Bearbeiten | Quelltext bearbeiten \]](#)

Die Verbindung wurde aufgrund der Erfahrungen aus der [Kubakrise](#) (14. bis 28. Oktober 1962) eingerichtet. Sie läuft über London, Kopenhagen, Stockholm, Helsinki und wurde am 30. August 1963^[1] eröffnet. Parallel dazu kam es zu einer Funkverbindung über [Tanger](#). 1966 folgte eine Verbindung der USA mit [Frankreich](#), 1967 mit [Großbritannien](#). Die Verbindungen sollen die Möglichkeit schaffen, eine Friedensgefährdung durch Irrtümer, Missverständnisse oder Verzögerungen im Kommunikationsweg zu verhindern.

Dass diese Gefahr weiterhin bestand, zeigt ein Vorfall vom 26. September 1983: Damals meldete die sowjetische Raketenabwehr fälschlich einen Angriff von US-Interkontinentalraketen auf die Sowjetunion (siehe den Artikel über [Stanislaw Jewgrafowitsch Petrow](#)).

Beide Seiten strebten eine höchstmögliche Sicherheit gegen Abhören oder Verfälschen der übermittelten Nachrichten an. Es kam die Verschlüsselungstechnik [One-Time-Pad](#) zum Einsatz – eine der wenigen bekannten Anwendungen des Verfahrens, das zwar absolute Sicherheit bietet, in der Praxis aufgrund des aufwendigen Schlüsselaustausches aber nur schwer durchführbar ist.



Ein Rotes Telefon aus der Zeit von [Jimmy Carter](#), das aber nie Bestandteil des heißen Drahtes, sondern wohl nur des amerikanischen [Defense Red Switch Networks](#) war.

SCHLÜSSEL NUR EINMAL VERWENDEN

One-Time Pad ist unsicher bei Wiederverwendung des gleichen Schlüssels

$$\begin{array}{ccccc} \boxed{m_0} & \oplus & \boxed{k} & = & \boxed{c_0} \\ & & & & \oplus \\ \boxed{m_1} & \oplus & \boxed{k} & = & \boxed{c_1} \end{array} \quad \rightarrow \quad \boxed{c_0 \oplus c_1 = m_0 \oplus m_1}$$

Lernen des XORs kann nützliche Informationen preisgeben:

- z.B. welche Bits von m_0 und m_1 gleich sind
- Wenn m_0 bekannt ist dann ist auch m_1 bekannt (und vice versa)

➔ **Moral:** Zur Verschlüsselung jeder Nachricht muss ein neuer zufälliger Schlüssel gewählt werden

NACHTEILE ONE-TIME PAD

1. Schlüssel ist **so lang** wie Nachricht

- Für große Mengen von Daten müssen lange zufällige Schlüssel gespeichert und ausgetauscht werden
- Gute Zufälligkeit zu erzeugen, ist sehr aufwendig

2. Schlüssel **kann nur einmal benutzt** werden:

- Kann etwas über Klartexte preisgeben

3. Sicherheit im **beschränkten** Angreifermodell

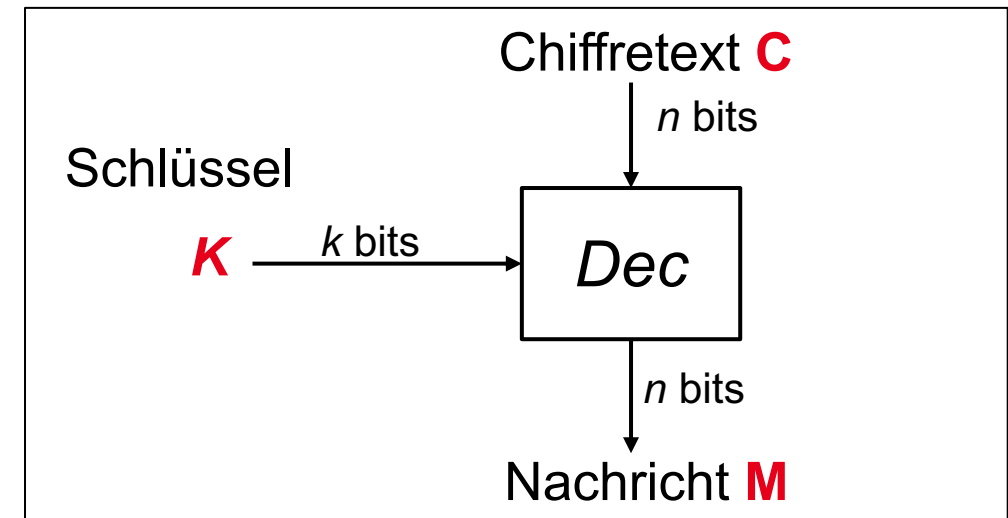
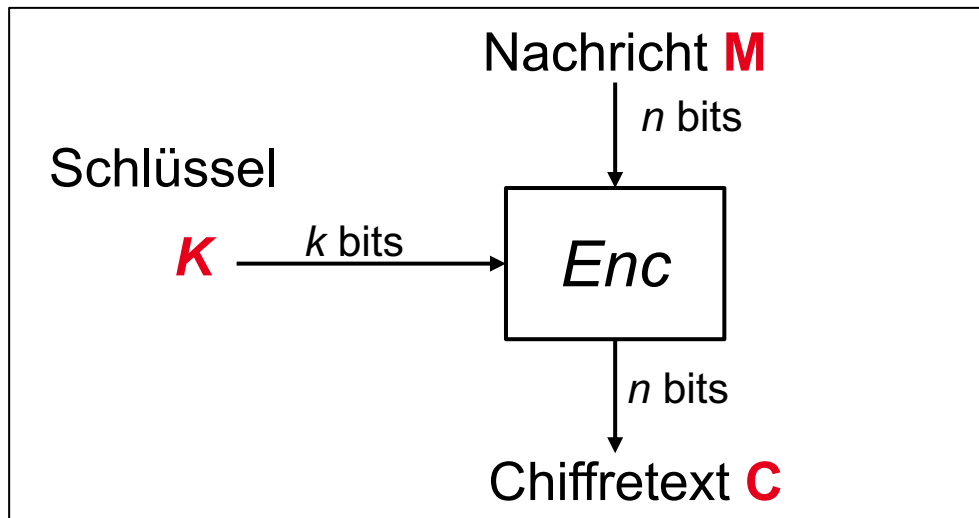
- Chiffretext-Only Angriffe



BLOCKCHIFFREN

BLOCKCHIFFRE

Verschlüsselung/Entschlüsselung von Nachrichten/Chiffretextblöcken mit **fixer Länge**

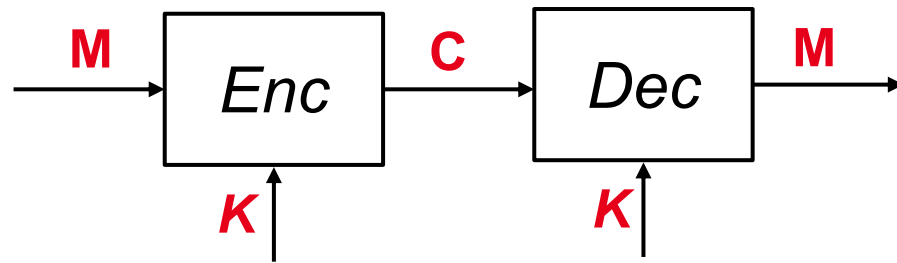


- **Blocklänge** $n = |m| = |c|$: häufig 64 – 128 Bits
- **Schlüssellänge** k : häufig 128 – 256 Bits

Wesentlicher Unterschied zu One-Time-Pad: Schlüssel kann wiederverwendet werden!

EIGENSCHAFTEN VON BLOCKCHIFFREN

- **Korrektheit:** Für jeder Nachricht **M** und jeden Schlüssel **K**



- **Effizienz:**

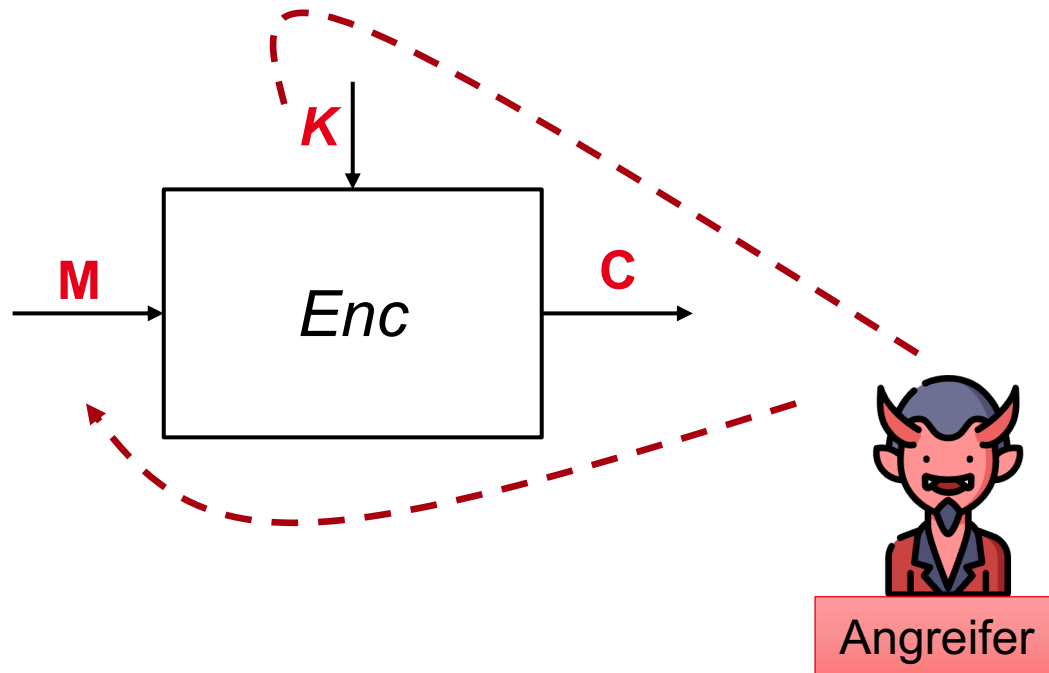
- **Enc(.)** und **Dec(.)** sollte in Mikrosekunden berechenbar sein: XOR, bit-shifting, Lookup Tabellen, ...
- Häufig Hardware Support auf modernen CPUs

- **Sicherheit:**

- Was soll ein Angreifer nicht können?

SICHERHEIT VON BLOCKCHIFFREN

Was soll ein Angreifer nicht können?

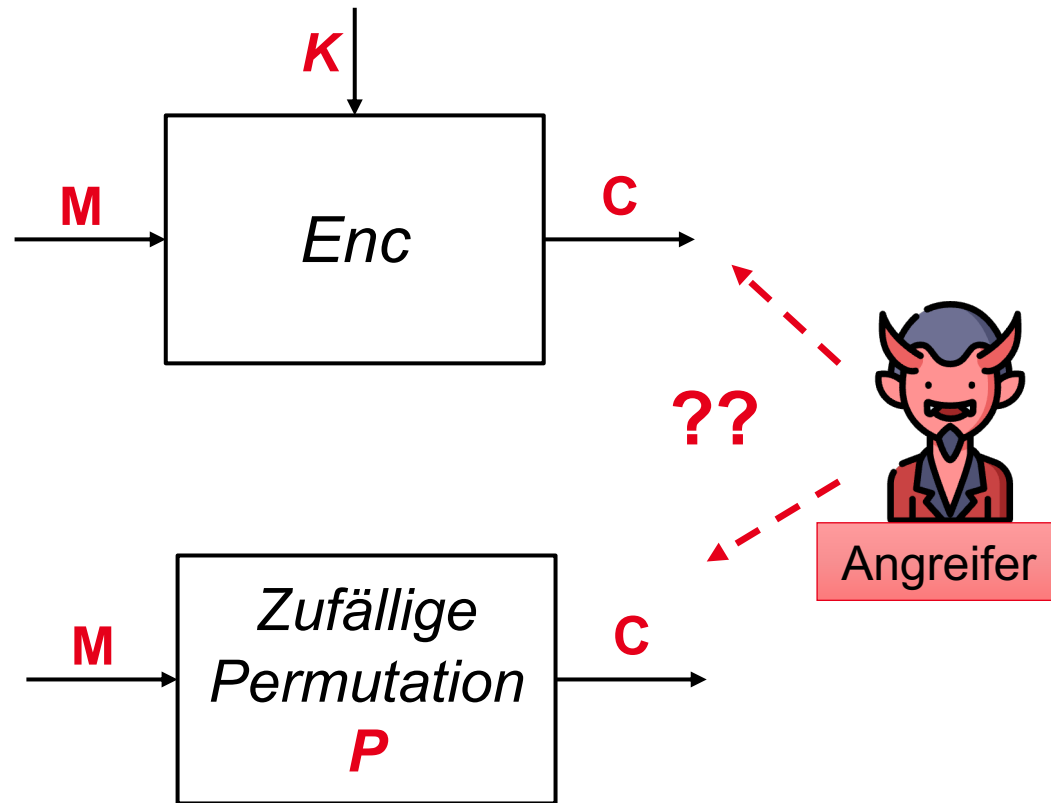


Aus **C** kann der Angreifer nicht...

- **M** lernen
- **K** lernen

SICHERHEIT VON BLOCKCHIFFREN

Was soll ein Angreifer nicht können?



Aus C kann der Angreifer nicht...

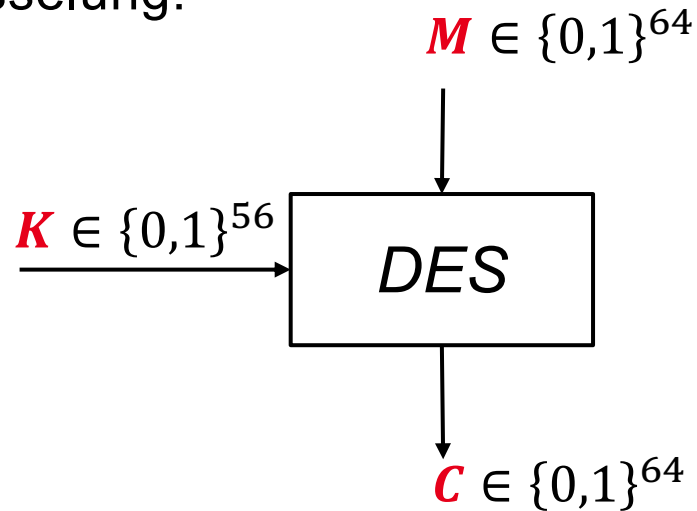
- M lernen
- K lernen

Formal: Angreifer kann nicht zwischen $Enc(.)$ und $P(.)$ unterscheiden.

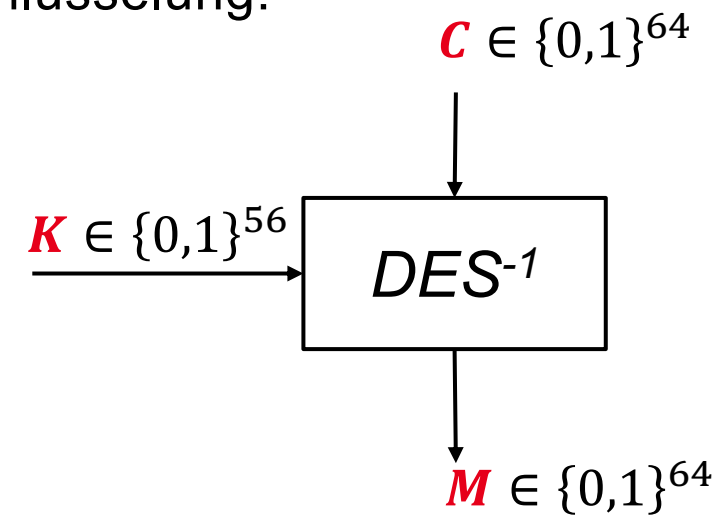
DATA ENCRYPTION STANDARD (DES)

Entwickelt von IBM im Auftrag des NIST Ende der 1970er

Verschlüsselung:



Entschlüsselung:



- Blocklänge **n = 64** Bits
- Schlüssellänge **k = 56** Bits

DES: ANGRIFFE



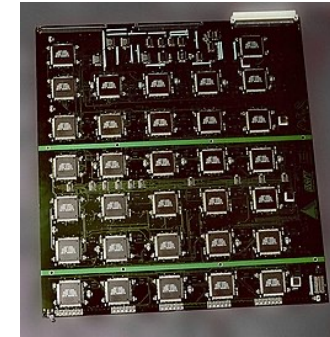
Theoretische Angriffe: **Differenzielle Kryptoanalyse** und **Lineare Kryptoanalyse**

- DES bietet Schutz gegen Differenzielle Kryptoanalyse

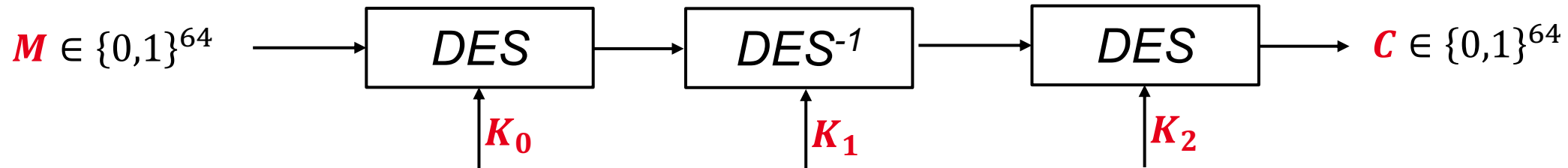


Hauptschwachpunkt: Kurzer Schlüssel (nur 56 Bits)

- Brute-Force Angriff ist möglich
- DES Cracker: Bricht DES in wenigen Tagen



Triple DES: Sicherheitsniveau ist 2 x 56 Bits (wegen Time-Memory-Trade-Off)



ADVANCED ENCRYPTION STANDARD (AES)

- Wettbewerb für **AES** wurde im **Januar 1997** von **US National Institute of Standards and Technology** (NIST) angekündigt.
- **15** Chiffren wurden eingereicht
- **5** Finalisten: **MARS, RC6, Rijndael, Serpent**, und **Twofish**
- **2. October, 2000**: **Rijandel** wurde als Gewinner gewählt.
- **26. November, 2001**: **AES** ist ein offizieller Standard geworden.
- Autoren: **Vincent Rijmen, Joan Daemen** (aus KU Leuven in Belgien)
- **Schlüssel-Größe**: **128, 192** or **256** bit, **Block-Größe**: **128** Bits (weitere werden von Rijndael unterstützt)
- **Gilt als ungebrochen**: In den USA Zulassung für höchste Geheimhaltungsstufe



Vincent Rijmen
(1970)



Joan Daemen
(1965)

BRUTE FORCE ANGRIFFE

Wie schwer ist ein Brute-Force-Angriff auf einen **128** Bit Schlüssel?

- Probiere **2^{128}** Schlüssel aus

Wie groß ist **2^{128}** ?

- **$2^{128} = 2^{10 \cdot 12.8} \approx 10^{3 \cdot 13} = 10^{39}$**

Angenommen wir haben Zugriff auf eine extrem leistungsstarke Hardware, die **10^9** (1 Milliarden) Schlüssel pro **Nanosekunde** testen kann

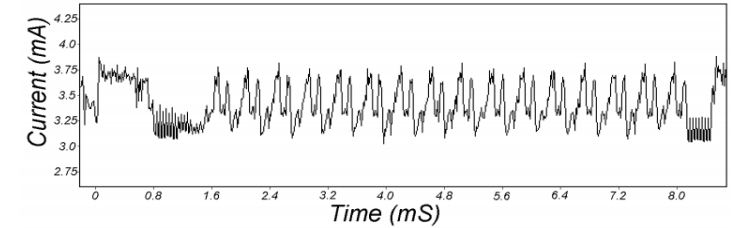
- D.h.: **10^{18}** Schlüssel pro Sekunde
- D.h.: Wir benötigen **$10^{39}/10^{18} = 10^{21}$** Sekunden
- **Das sind ca. 30 Billionen Jahre**

Takeaway: Brute-Force Angriffe auf moderne Chiffren mit Schlüsseln \geq **128** Bits unmöglich

ANGRIFFE IN DER PRAXIS

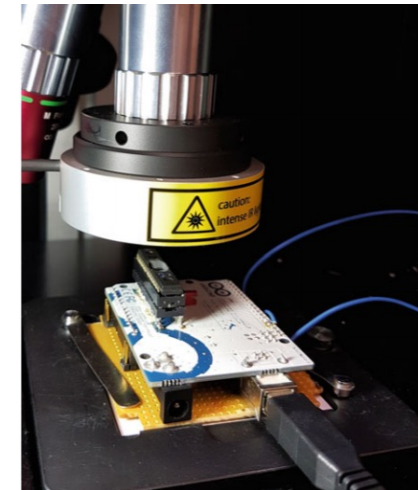
1. Seiten-Kanal-Angriffe

- **Beobachte** das Gerät bei Ent-/Verschlüsselung
 - Messe die **Zeit**, die für kryptographische Operationen benötigt wird.
 - Messe den **Stromverbrauch**



2. Fehlerangriffe

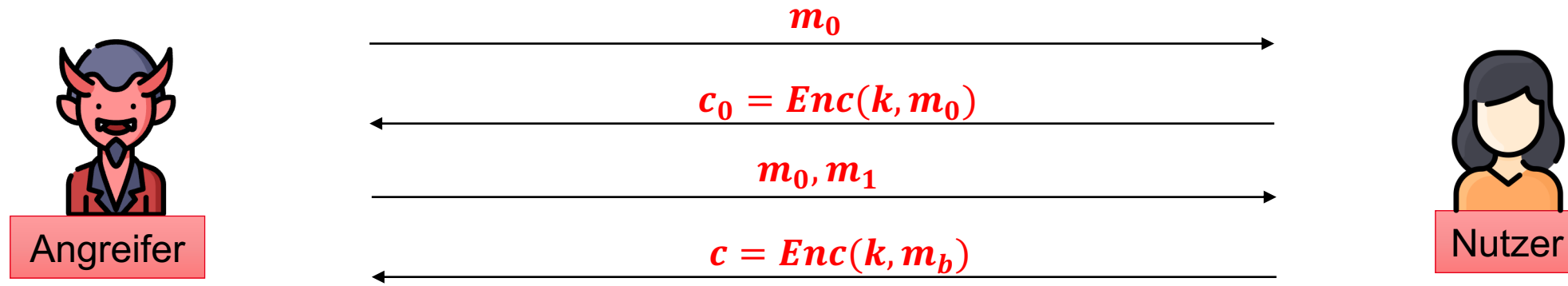
- **Füge Fehler** (z.B. mit einem Laserstrahl) in die Berechnung ein
- Beobachte die **Veränderungen** im Verhalten der **Ein-/Ausgabe**
- z.B. Berechnungsfehler in den letzten paar Runden von DES erlauben es, den Schlüssel **K** zu berechnen



BLOCKCHIFFREN IM EINSATZ

Blockchiffren besitzen zwei Probleme bei direktem Einsatz als Verschlüsselung:

- Nicht IND-CPA sicher:



Gib **0** aus falls $c = c_0$; ansonsten **1**

Merke: Deterministische Verschlüsselung kann nicht IND-CPA sicher sein

- Nicht möglich Nachrichten beliebiger Länge zu entschlüsseln



MODES OF OPERATION

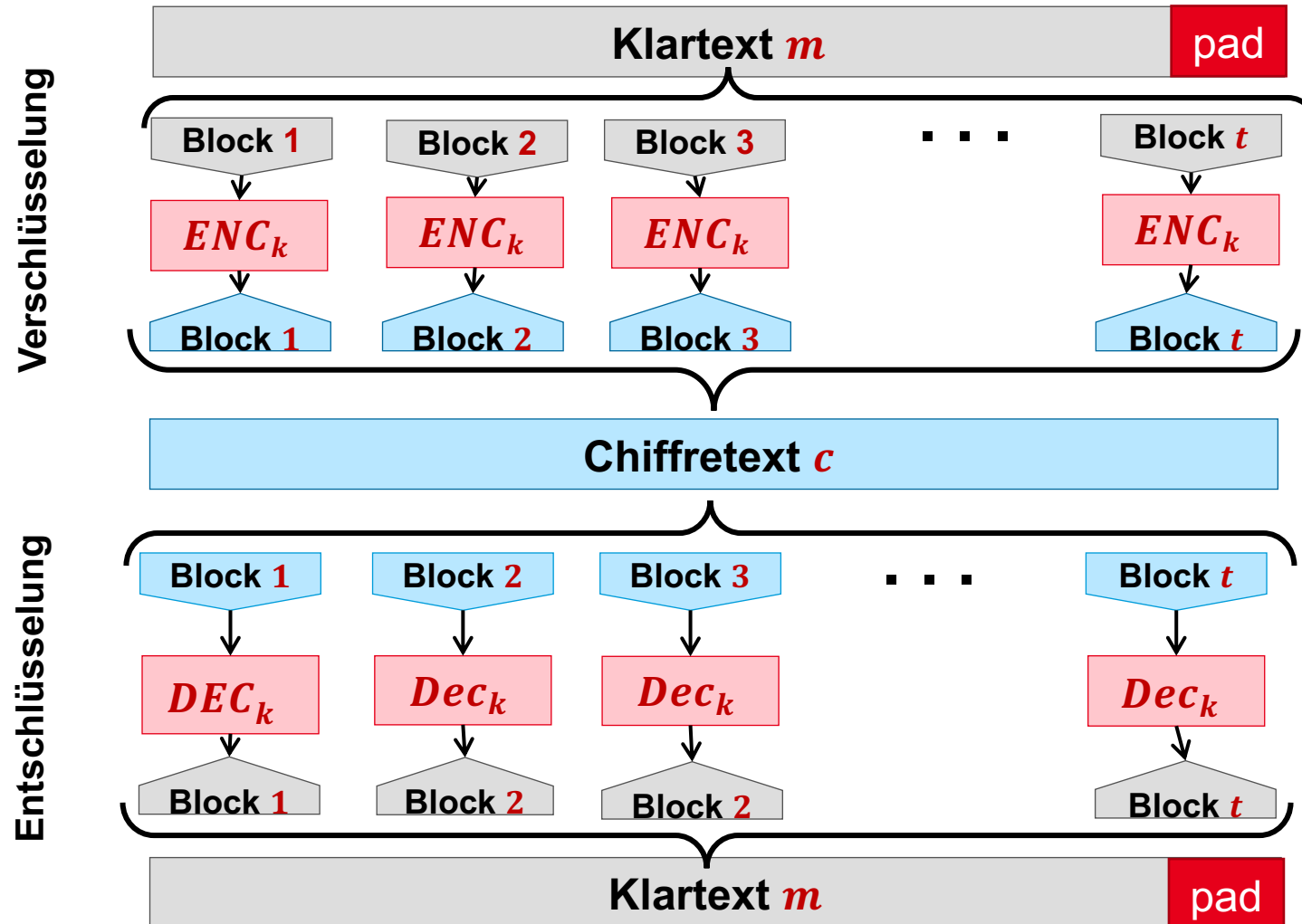
MODES OF OPERATION

Ziel: Verschlüsselung von Nachrichten mit **beliebiger Länge**

- Ohne Vergrößerung des Chiffretexts (im Vergleich zum Klartext)
 - Blockchiffren **können nicht direkt für die Verschlüsselung benutzt werden.**
 - Sie werden immer in bestimmten “**Modes of Operations**” benutzt
1. **Electronic Codebook (ECB)** Modus ← **nicht sicher,**
 2. **Cipher-Block Chaining (CBC)** Modus,
 3. **Counter (CTR)** Modus,

...

ELECTRONIC CODE BOOK (ECB) MODUS



Wenn $|m|$ kein Vielfaches der **Blocklänge**, dann muss der Klartext um ein Padding „**pad**“ ergänzt werden

SCHWÄCHE DES ECB MODUS

ECB Modus ist nicht sicher, da **deterministisch**!

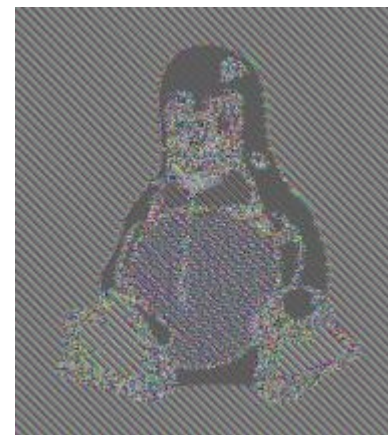
$$M_1 = M_2 \rightarrow C_1 = C_2$$

Bilddatei Format

W	W	W
W	B	B
W	B	B
W	Y	Y
B	W	...



ECB



Quelle: Wikipedia

Verschlüsselte Bilddatei

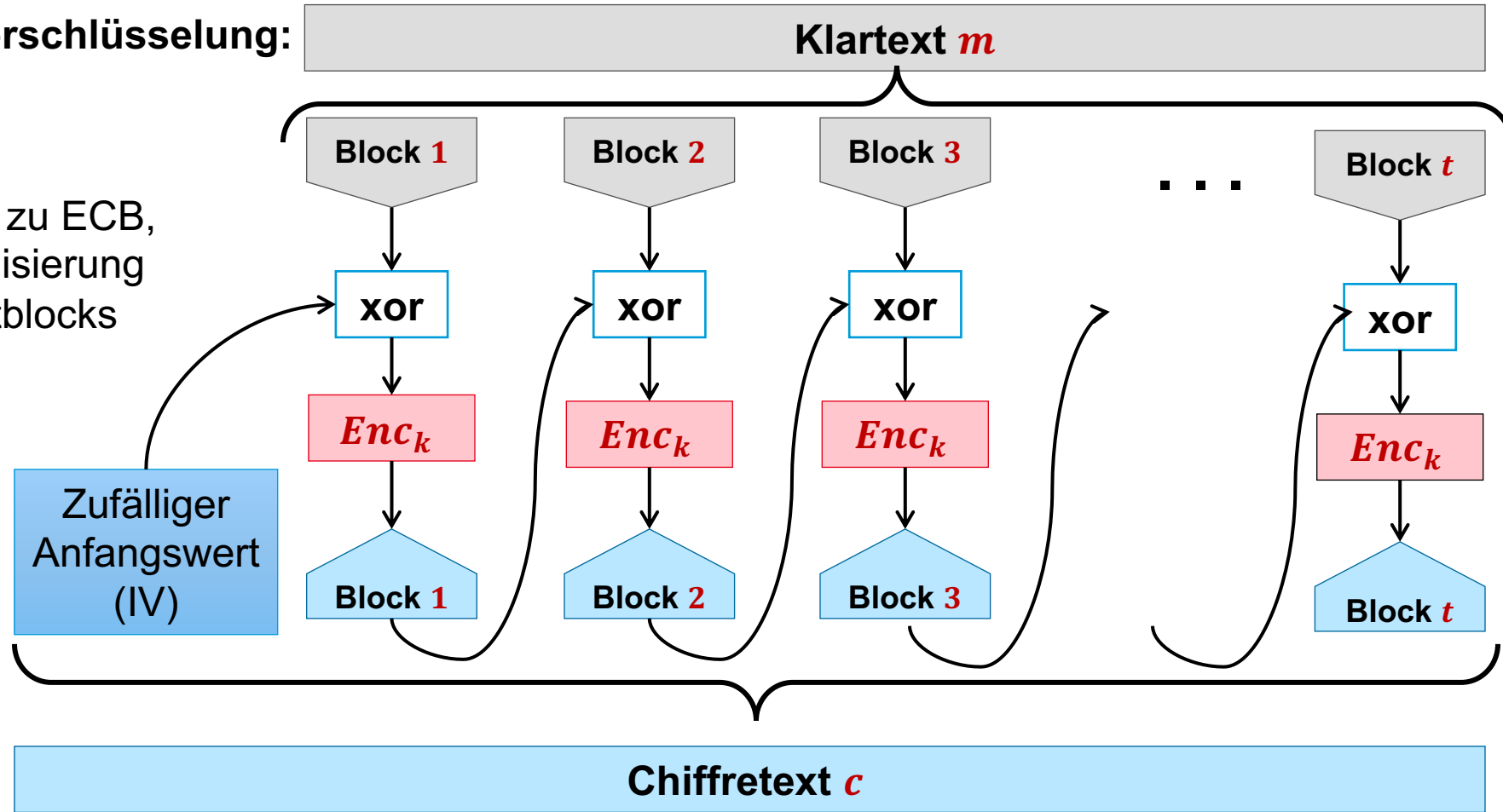
E(k,W)	E(k,W)	E(k,W)
E(k,W)	E(k,B)	E(k,B)
E(k,W)	E(k,B)	E(k,B)
E(k,W)	E(k,Y)	E(k,Y)
E(k,B)	E(k,W)	...

CIPHER BLOCK CHAINING (CBC) MODUS



Verschlüsselung:

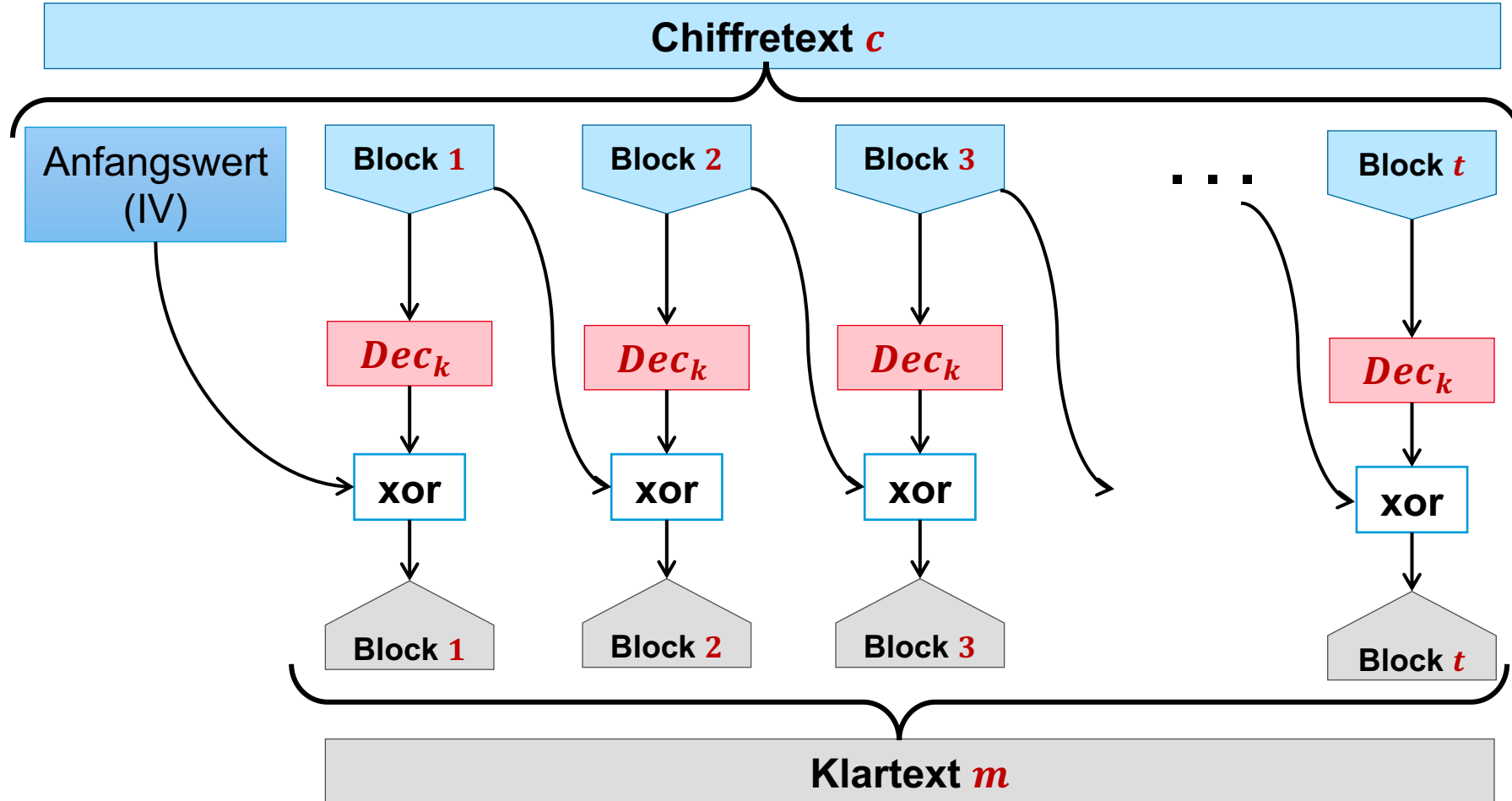
Idee: Ähnlich zu ECB,
aber Randomisierung
jedes Klartextblocks



CIPHER BLOCK CHAINING (CBC) MODUS



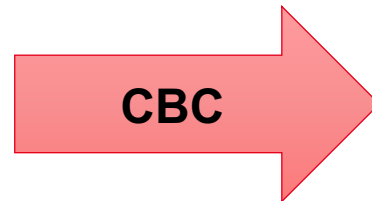
Entschlüsselung:



CBC MODUS: SICHERHEIT

Was passiert, wenn zweimal die gleiche Nachricht verschlüsselt wird?

$M_1 = M_2 \rightarrow C_1 \neq C_2$ (wegen zufälliger IV)



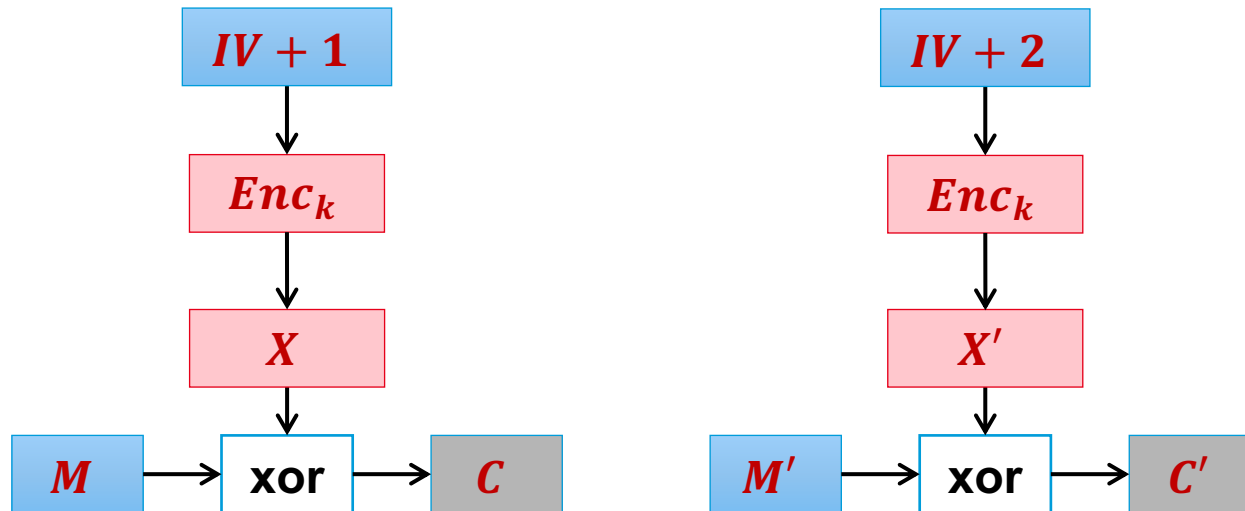
CBC ist **sicher**, wenn richtig verwendet!

Achtung: Probleme mit Padding sind häufig in der Praxis!

COUNTER MODUS (CTR)

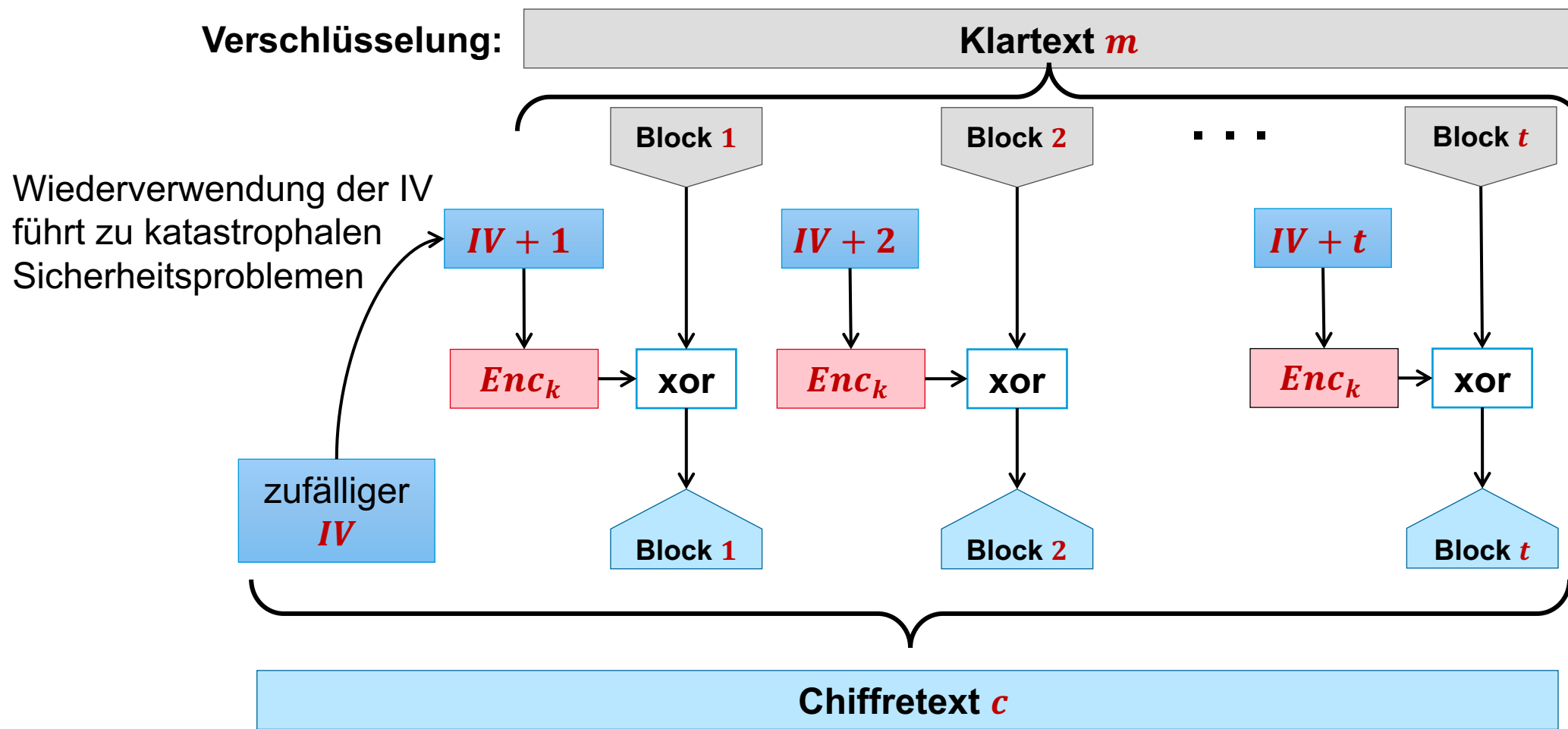
Wiederholung: One-Time-Pad

- Wenn Schlüssel **X** zufällig und nicht wiederverwendet wird dann sicher



Idee: Nutze Ausgabe der Blockchiffre als One-Time-Pad

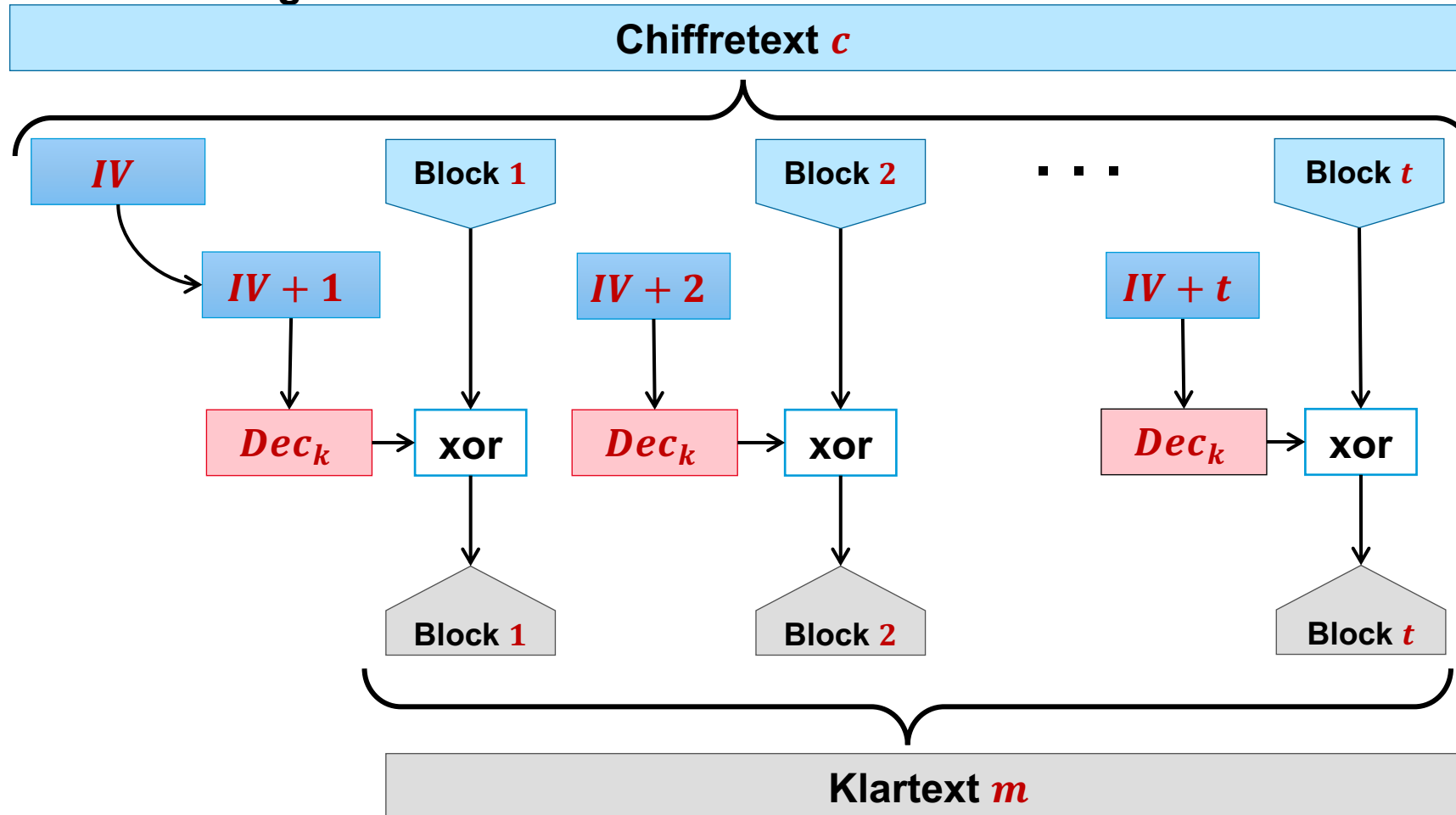
COUNTER MODUS (CTR)



COUNTER MODUS (CTR)



Entschlüsselung:



VIELE WEITERE MODI

- Cipher Feedback Mode (CFB)
- Output Feedback Mode (OFB)
- Galois Counter Mode
- Counter Mode with CBC-MAC (CCM)

} Authentifizierte Modi (später mehr dazu)