Übung zur Vorlesung „Sicherheit“
Übung 2

Thomas Agrikola
Thomas.Agrikola@kit.edu

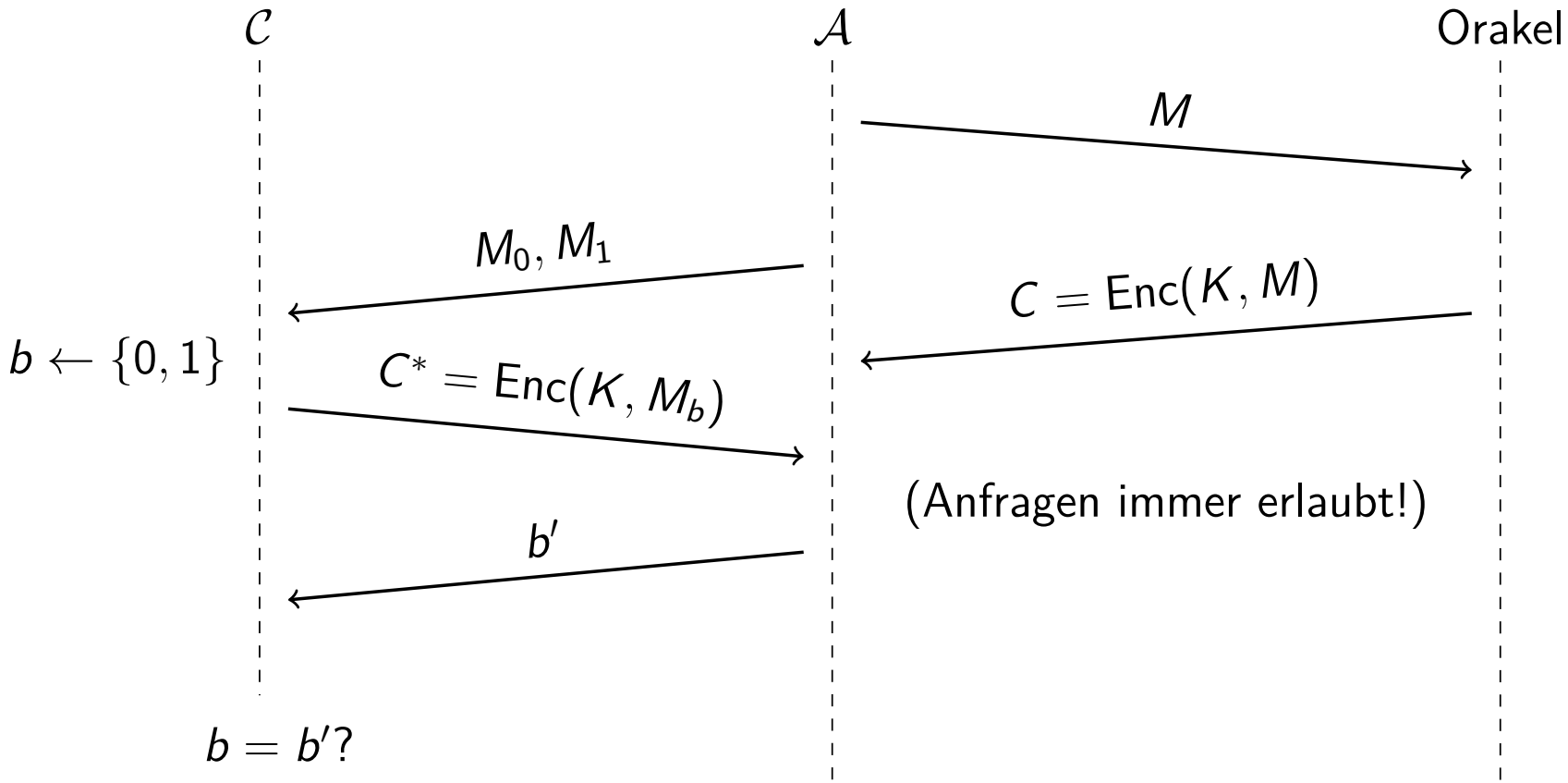
04.05.2017

Organisatorisches

- Nachklausur:
  - Teilnahme an Hauptklausur *nicht* notwendig um an Nachklausur teilnehmen zu können
  - Termin Hauptklausur: 02.08.2017, 14:30
  - Termin Nachklausur: 06.10.2017, 14:00

Wdh.: IND-CPA für symm. Verschlüsselung

- Herausforderer $\mathcal{C}$ wählt Schlüssel K zufällig.
- $\mathcal{C}$ stellt $\mathrm{Enc}(K, \cdot)$-Orakel für $\mathcal{A}$ bereit.



Socrative: IND-CPA

`https://b.socrative.com/login/student/`

Room: `SICHERHEIT`

- App um Quiz durchzuführen
- Zugang durch Browser oder App
- Als Quizteilnehmer kein Account notwendig.

ÜB 2 – Aufgabe 1

Zeigen Sie: Der CBC-Modus kann nicht IND-CPA-sicher sein, wenn der Initialisierungsvektor IV

1. fest und für jeden Verschlüsselungsvorgang gleich gewählt, oder
2. ausgehend von einer fixen Wahl bei jeder Verschlüsselung um 1 hochgezählt wird.

ÜB2 – Aufgabe 1: *IV* fest

Erinnerung: CBC-Modus

- $C_0 = IV$,
- $C_i = \mathrm{E}(K, M_i \oplus C_{i-1})$.

Beobachtung: *IV* fest $\rightarrow$ CBC-Modus deterministisch.

ÜB2 – Aufgabe 1: *IV* fest

Konstruiere folgenden Angreifer $\mathcal{A}$:

- $\mathcal{A}$ wählt zwei gleichlange Nachrichten $M_0 \neq M_1$,
- gibt M_0 und M_1 an den Challenger $\mathcal{C}$ aus
- und erhält ein Challenge-Chiffrat C^*.
  - C^* ist Chiffrat von M_0 oder M_1.
  - $\mathcal{A}$ muss entscheiden, was in C^* verschlüsselt ist!
- $\mathcal{A}$ gibt M_0 an sein Verschlüsselungsorakel und erhält $C_0 = \mathrm{Enc}(K, M_0)$.
- $\mathcal{A}$ überprüft, ob $C^* = C_0$. Wenn ja, gibt er 0 aus, sonst 1.

$\Rightarrow \mathcal{A}$ gewinnt **immer**!

ÜB2 – Aufgabe 1: IV wird hochgezählt

Sei

- ℓ die Blocklänge und
- $IV_1 \in \{0,1\}^\ell$ der erste Initialisierungsvektor.

Konstruiere folgenden Angreifer $\mathcal{A}$:

- $\mathcal{A}$ schickt $M_0 = 0^\ell$ ans Orakel und erhält

$$\begin{aligned} C = \mathsf{Enc}(K, M_0) &= (IV_1, \mathsf{E}(K, M_0 \oplus IV_1)) \\ &= (IV_1, \underbrace{\mathsf{E}(K, IV_1)}_{=:X}). \end{aligned}$$

- $\mathcal{A}$ merkt sich $X := \mathsf{E}(K, IV_1)$ und IV_1 und berechnet $IV_2 = IV_1 + 1$.

ÜB2 – Aufgabe 1: IV wird hochgezählt

- $\mathcal{A}$ setzt $M_1 = IV_1 \oplus IV_2$.
- Er schickt die Nachrichten M_0, M_1 an den Challenger und erhält $C^* = (C_1^*, C_2^*) = (IV_2, \mathrm{E}(K, M_b \oplus IV_2))$.
- Wurde M_1 verschlüsselt, so gilt:

$$C_2^* = \mathrm{E}(K, M_1 \oplus IV_2) = \mathrm{E}(K, IV_1 \oplus IV_2 \oplus IV_2)$$
$$= \underbrace{\mathrm{E}(K, IV_1)}_{=X}.$$

- Falls $C_2^* = X$ gilt, so gibt $\mathcal{A}$ die Ausgabe 1 aus, sonst 0.

$\Rightarrow \mathcal{A}$ gewinnt **immer**!

ÜB2 – Aufgabe 1 – Fazit

- Nicht deterministische Verschlüsselung **notwendig** für IND-CPA-Sicherheit
  - ... aber nicht hinreichend!

ÜB2 – Aufgabe 2

Es sei $\mathsf{SKE} = (\mathsf{Enc}, \mathsf{Dec})$ ein IND-CPA-sicheres, symmetrisches Verschlüsselungsverfahren.

Wir konstruieren daraus zwei neue Verfahren.

ÜB2 – Aufgabe 2 (1.) (a)

Betrachte SKE′ mit

- $\mathsf{Enc}'(K, M) := \mathsf{Enc}(K, \mathsf{Enc}(K, M))$,
- $\mathsf{Dec}'(K, C) := \mathsf{Dec}(K, \mathsf{Dec}(K, C))$.

Intuitiv: verschlüssle Nachrichten zweifach.

(a) Korrektheit:

$$\begin{aligned}\mathsf{Dec}'(K, \mathsf{Enc}'(K, M)) &= \mathsf{Dec}(K, \mathsf{Dec}(K, \mathsf{Enc}(K, \mathsf{Enc}(K, M)))) \\ &= \mathsf{Dec}(K, \mathsf{Enc}(K, M)) \\ &= M \quad \checkmark\end{aligned}$$

ÜB2 – Aufgabe 2 (1.) (a)

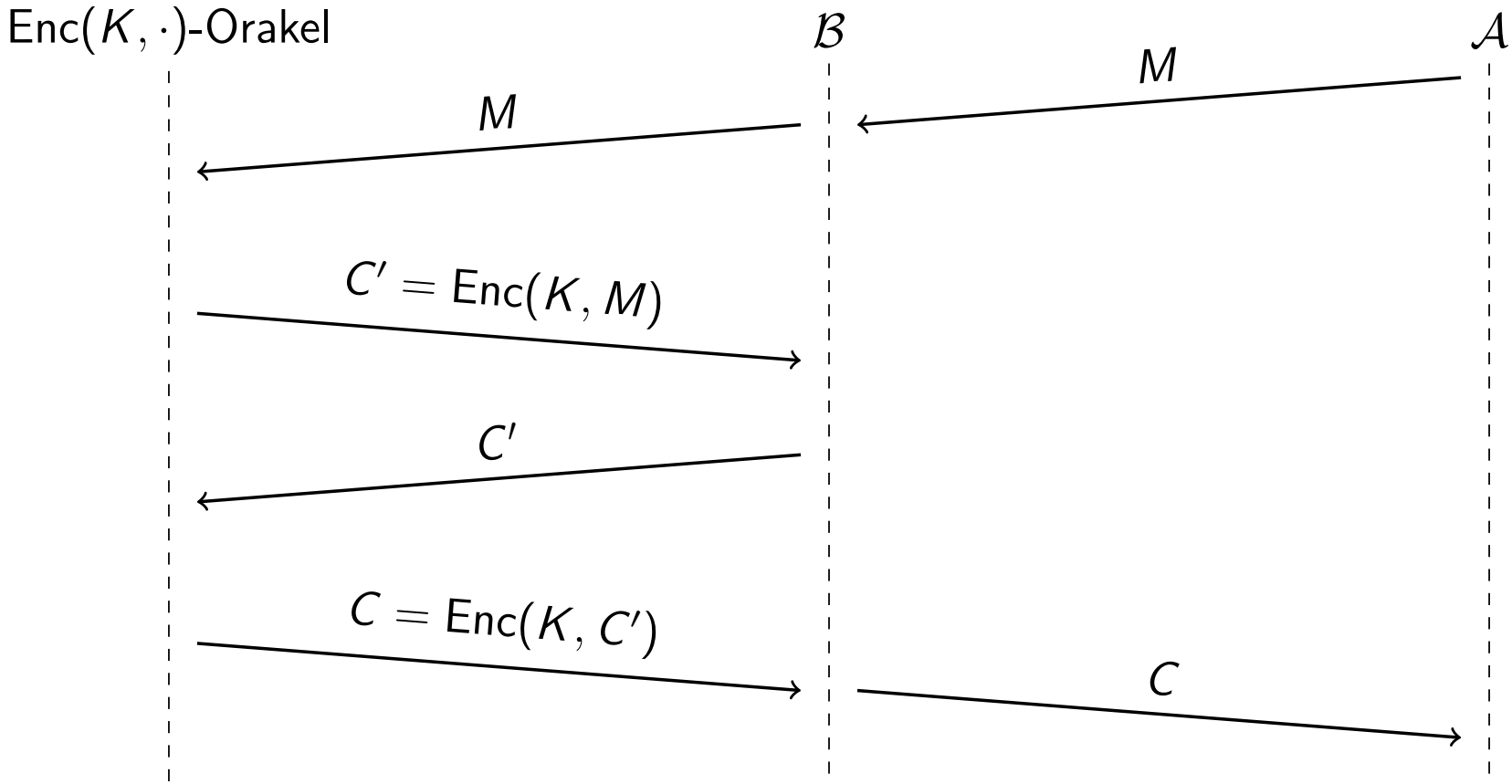
(b) IND-CPA-Sicherheit:
Strategie: **Reduktion** auf IND-CPA-Sicherheit von SKE

- Annahme: $\exists$ effiz. "erfolgreicher" Angreifer $\mathcal{A}$ auf SKE$'$.
- Konstruiere aus $\mathcal{A}$ einen "erfolgreichen" IND-CPA-Angreifer $\mathcal{B}$ auf SKE.
- Aber: $\mathcal{B}$ kann nicht existieren, da SKE IND-CPA-sicher ist!

$\Rightarrow \mathcal{A}$ kann nicht existieren.

ÜB2 – Aufgabe 2 (1.) (b)

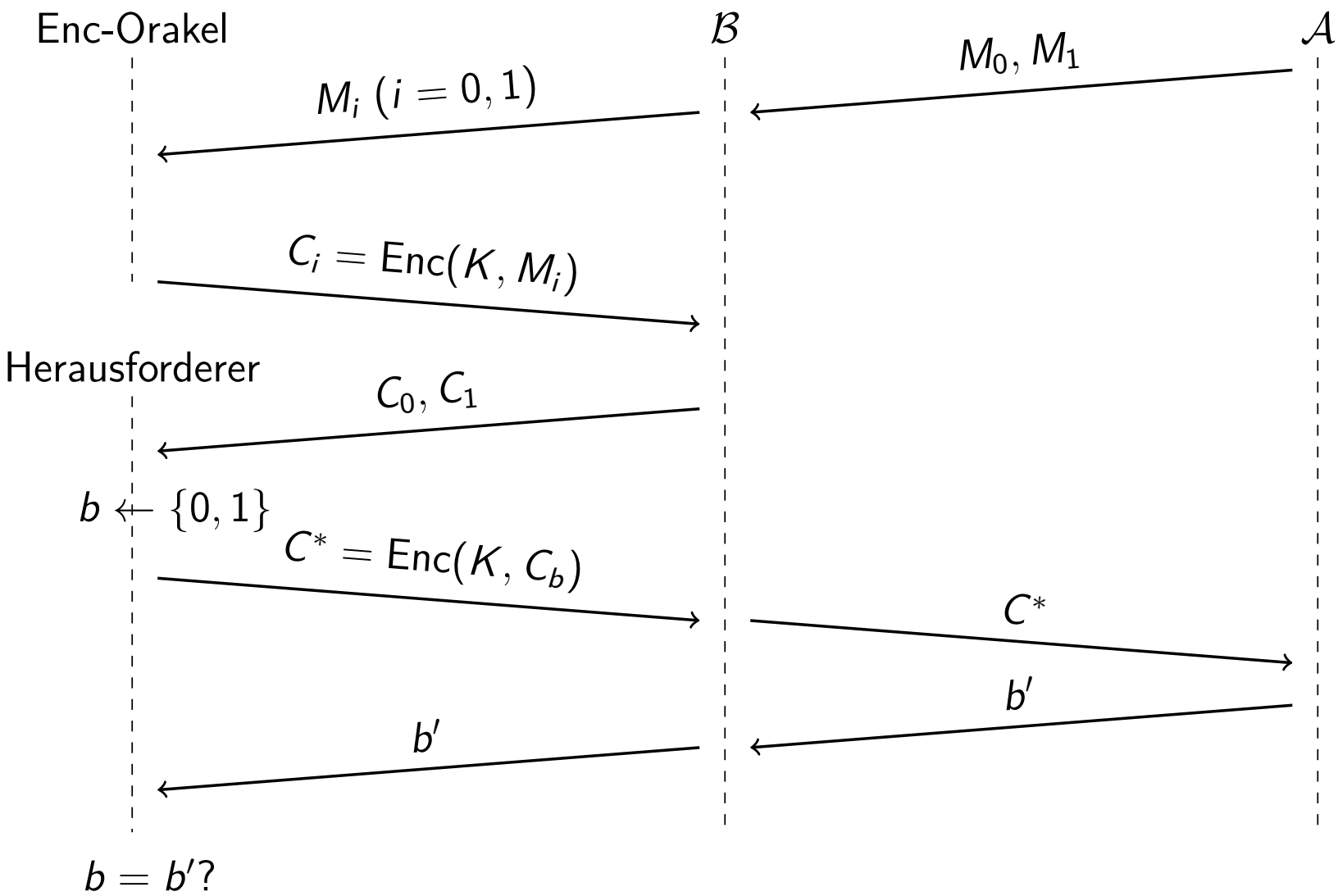
Simulation des Enc'-Orakels:



Beobachtung: $C = \mathsf{Enc}(K, \mathsf{Enc}(K, M)) = \mathsf{Enc}'(K, M)$!

ÜB2 – Aufgabe 2 (1.) (b)

Challenge:



ÜB2 – Aufgabe 2 (1.) (b)

- $\mathcal{B}$ simuliert das IND-CPA-Spiel für $\mathcal{A}$ perfekt.
- $\mathcal{B}$ gewinnt $\Leftrightarrow$ $\mathcal{A}$ gewinnt.
  $\Rightarrow$ $\mathcal{B}$ ist "erfolgreicher" Angreifer auf SKE

$\Rightarrow$ Widerspruch zur IND-CPA-Sicherheit von SKE.
$\Rightarrow$ SKE$'$ ist IND-CPA-sicher!

ÜB2 – Aufgabe 2 (1.) – Fazit

Doppelte Verschlüsselung:

- ... ist auch IND-CPA-sicher, wenn das Verfahren bereits IND-CPA-sicher war.
- Vorteil/Sicherheitsgewinn?
- Evtl. nicht sinnvoll, falls bereits grundlegendes Verfahren unsicher
  - Vergleiche mit 2DES

ÜB2 – Aufgabe 2 (2.) (a)

Betrachte SKE^* mit

- $\mathsf{Enc}^*(K, M) := (M_{(0)}, \mathsf{Enc}(K, M))$ $(= (C_1, C_2))$,
- $\mathsf{Dec}^*(K, C) := \mathsf{Dec}(K, C_2)$.

$M_{(0)}$ ist das niederwertigste Bit von M.

(a) Korrektheit:

$$\begin{aligned} \mathsf{Dec}^*(K, C) &= \mathsf{Dec}(K, C_2) \\ &= \mathsf{Dec}(K, \mathsf{Enc}(K, M)) \\ &= M \quad \checkmark \end{aligned}$$

ÜB2 – Aufgabe 2 (2.) (b)

(b) IND-CPA-Sicherheit:

- Angreifer $\mathcal{A}$ wählt M_0, M_1 mit $M_{0,(0)} \neq M_{1,(0)}$ (niederwertige Bits verschieden).
- Das Challenge-Chiffrat hat dann folgende Form: $C^* = (\underbrace{C_1^*}_{M_{b,(0)}}, C_2^*)$.
- $\mathcal{A}$ kann eindeutig entscheiden, welche Nachricht verschlüsselt wurde.
- SKE* **nicht** IND-CPA-sicher.

ÜB2 – Aufgabe 2 (2.) – Fazit

Damit ein Verschlüsselungsschema IND-CPA-sicher sein kann, darf ...

- ... nicht mal ein Bit Information über den Klartext „leaken“.
- ... kein Teil des Chiffrats eindeutig deterministisch vom Klartext abhängen.

IND-CPA & Praxis I

Frage: „Anwendung“ von Sicherheit gegen Chosen-Plaintext-Angriffe?

- IND-CPA = Möglichkeit, Chosen-Plaintext-Angriffe theoretisch abzubilden
- Deckt „Lauschangriffe“ komplett ab
- Deckt Known-Plaintext-Angriffe komplett ab
  - Known Plaintext: $\mathcal{A}$ kennt Klartext zu Chiffrat (oder Teile davon), kann ihn aber nicht selbst wählen.
  - Viele historische Verfahren anfällig dafür, z.B. Vigenère, Enigma, ...
  - Heutzutage: Headerdaten, bekannte Formatierung, bekanntes Dateiformat...

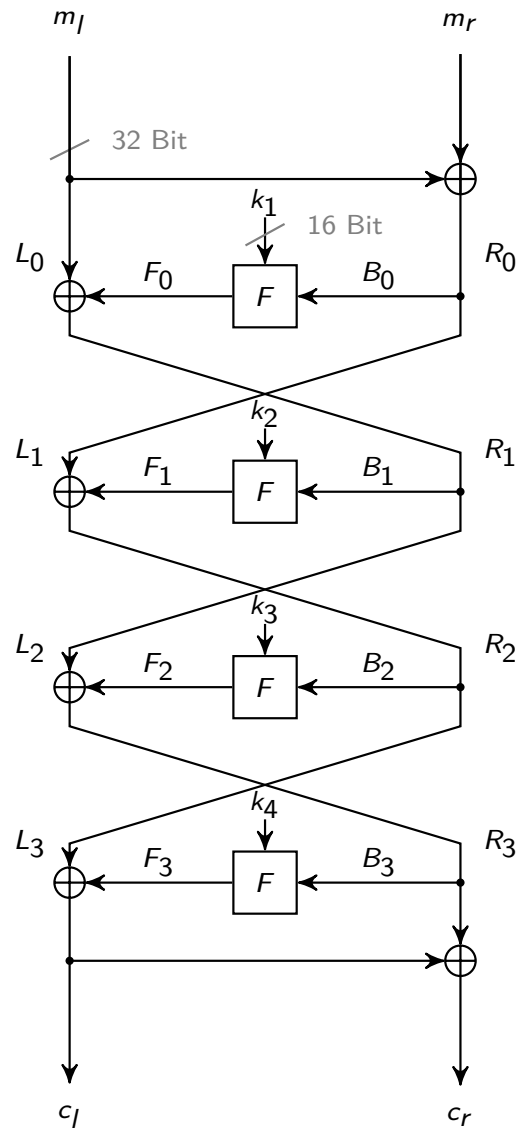
IND-CPA & Praxis II

- Chosen-Plaintext & asymmetrische Verfahren
  - Public Key zum Verschlüsseln ist öffentlich...
  - ... Angriff damit sehr realistisch
- Chosen-Plaintext & symmetrische Verfahren
  - Schlüssel geheim – wie Chosen-Plaintext-Angriff durchführen?
  - evtl. keine komplette Kontrolle über Klartext, aber zumindest Teile
  - „kreative“ Wege aus der Geschichte (z.B. 2. Weltkrieg)
  - Schadsoftware & Softwarefehler
  - Benutzeranfragen
  - ...

ÜB2 – Aufgabe 3: Lineare Kryptoanalyse

FEAL4:

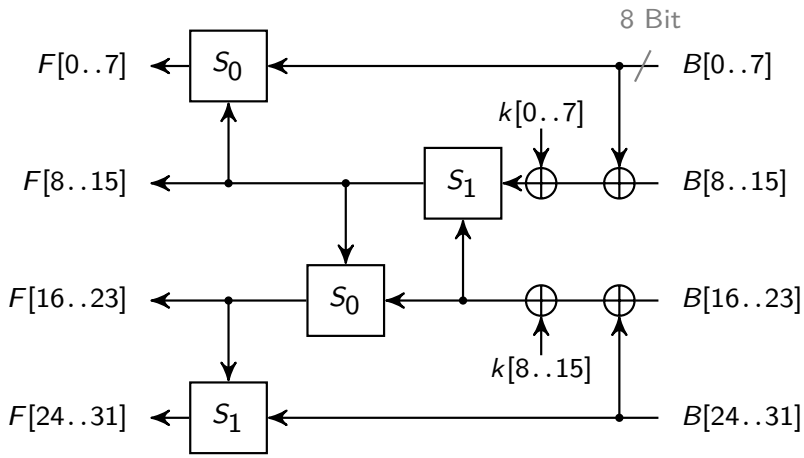
- Feistelstruktur mit 4 Runden
- Nachrichtenlänge: 64 Bit, L und R jeweils 32 Bit
- Eingabe- und Ausgabelänge von F: 32 Bit
- Runden-Schlüssel $K_1, \ldots, K_4$: 16 Bit



ÜB2 – Aufgabe 3: Lineare Kryptoanalyse

Die F-Funktion:

- jede "Leitung": 8 Bit
- S-Boxen:
  $S_i(a, b) :=$
  $rot2((a + b + i) \mod 256)$



ÜB2 – Aufgabe 3: Lineare Kryptoanalyse

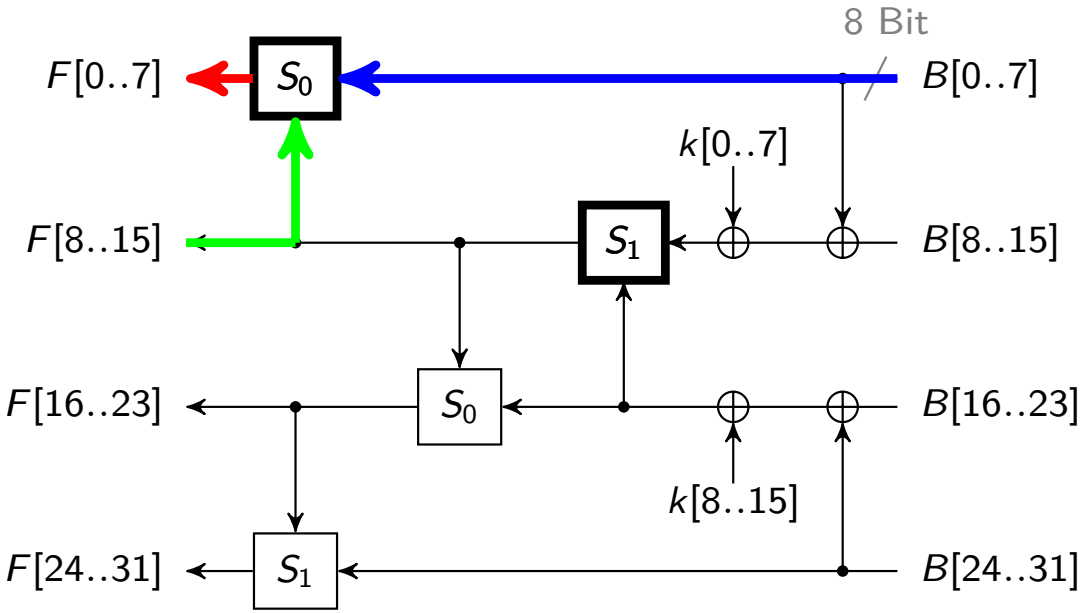
- **Beobachtung:** S-Boxen verhalten sich in gewisser Weise linear
  - niederwertiges Bit von $(a + b) \mod 256$ entspricht der Summe der niederwertigen Bits von a und b
    (es kann kein Übertrag auftreten)
  - nach Rotation um 2 nach links:

$$S_0(a, b)[2] = a[0] \oplus b[0]$$
$$S_1(a, b)[2] = a[0] \oplus b[0] \oplus 1$$

- Benutze diese Beobachtung um lineare Gleichungen über die F-Funktion zu erhalten

ÜB2 – Aufgabe 3(a)

Welche Gleichungen liefern die S-Boxen?



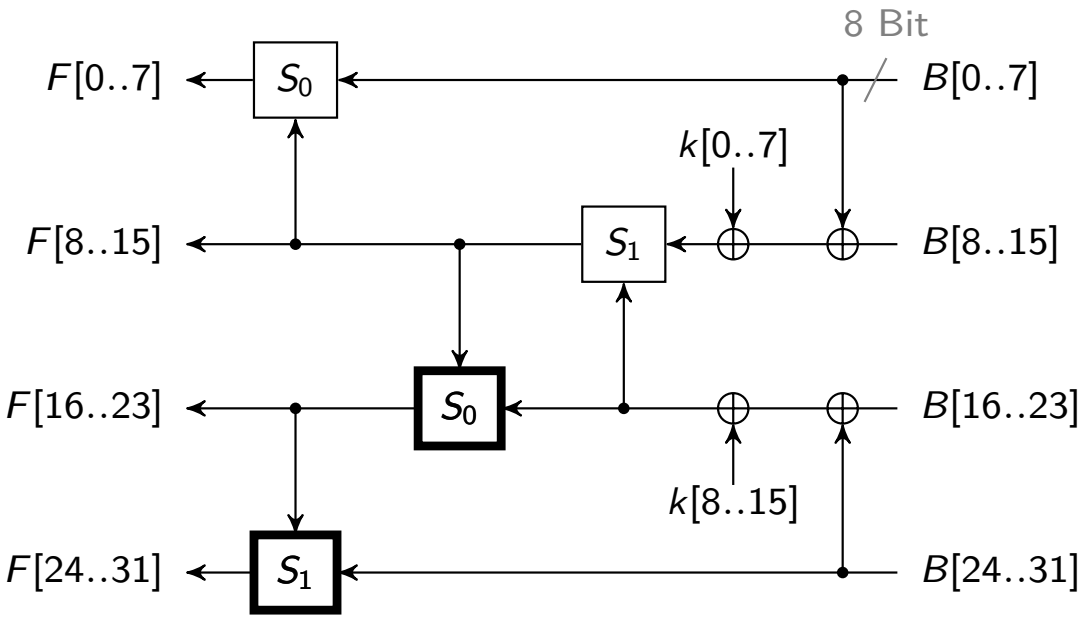
$$S_0(a,b)[2] = a[0] \oplus b[0]$$
$$S_1(a,b)[2] = a[0] \oplus b[0] \oplus 1$$

1. S-Box: $F[2] = B[0] \oplus F[8]$

2. S-Box: $F[10] = 1 \oplus (B[8] \oplus B[0] \oplus K[0]) \oplus (B[16] \oplus B[24] \oplus K[8])$

ÜB2 – Aufgabe 3(a)

Welche Gleichungen liefern die S-Boxen?



$$S_0(a,b)[2] = a[0] \oplus b[0]$$
$$S_1(a,b)[2] = a[0] \oplus b[0] \oplus 1$$

3. S-Box: $F[18] = F[8] \oplus (B[16] \oplus B[24] \oplus K[8])$

4. S-Box: $F[26] = 1 \oplus F[16] \oplus B[24]$

ÜB2 – Aufgabe 3(b)

Wie kann man eine 3-Runden Charakteristik ableiten?

$$F_0[2] = B_0[0] \oplus F_0[8]$$

$$m_l[0] \oplus m_r[0]$$

$$m_l[2] \oplus R_1[2]$$

$$R_3[2] \oplus F_2[2]$$

$$B_2[0] \oplus F_2[8]$$

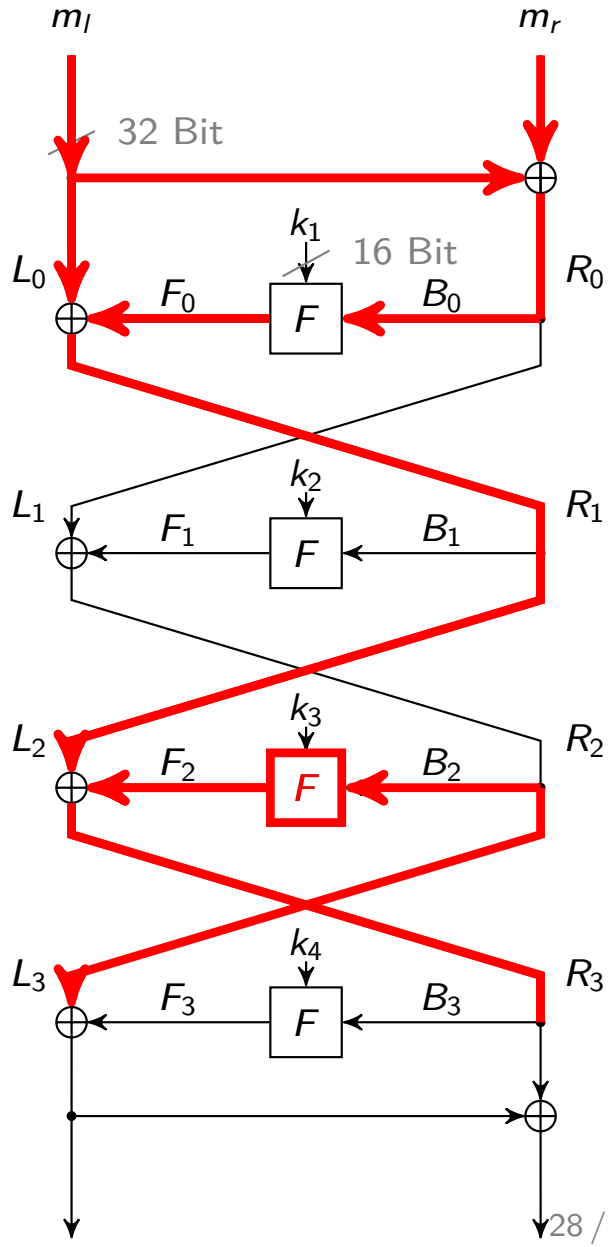
$$R_2[0]$$

$$L_3[0]$$

$$m_l[8] \oplus R_1[8]$$

$$R_3[8] \oplus F_2[8]$$

$$\Rightarrow m_l[0, 2, 8] \oplus m_r[0] \oplus R_3[2, 8] \oplus L_3[0] = 0.$$



ÜB2 – Aufgabe 3(b)

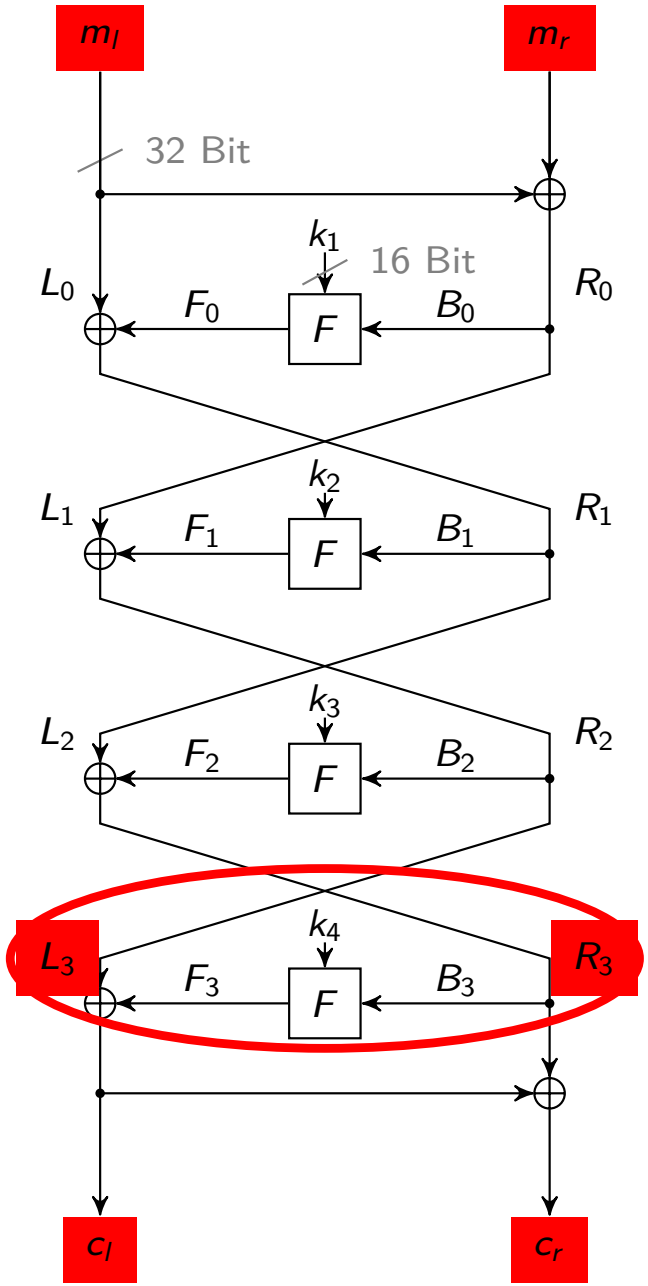
Die dritte Charakteristik $F[18] = F[8] \oplus B[16] \oplus B[24] \oplus K[8]$ dehnt sich analog aus zu:

$$m_l[16, 24, 8, 18] \oplus m_r[16, 24] \oplus R_3[8, 18] \oplus L_3[16, 24] = \underbrace{K_1[8] \oplus K_3[8]}_{\text{konstant}}$$

ÜB2 – Aufgabe 3(c)

Wie gehen wir weiter vor?

- Gegeben: Klartext-Chiffrat-Paare
- Vollständige Suche über alle 2^{16} Schlüssel K_4:
  - Entschlüssle Chiffrat "teilweise" mit K_4
    $\rightsquigarrow$ erhalte jeweils L_3 und R_3
  - Prüfe für jedes K_4, ob alle Charakteristiken für alle gegebenen Klartext-Chiffrat-Paare gelten
  - falls ja: Schlüsselkandidat
  - falls nein: K_4 definitiv falsch



ÜB2 – Aufgabe 3(c)

Nachdem wir K_4 kennen:

- Angriff auf K_1 analog wie auf K_4
- sind K_1 und K_4 bekannt, können K_2 und K_3 jeweils durch eine vollständige Suche über K_2 bzw. K_3 (Aufwand jeweils 2^{16}) gewunden werden

ÜB2 – Aufgabe 3(d)

Wie viele Klartext-Chiffrat-Paare reichen aus?

- Sei K_4 ein falscher Teilschlüssel.
- Idealerweise ist die Wahrscheinlichkeit, dass eine Charakteristik für n Klartext-Chiffrat-Paare trotzdem konstant ist, höchstens $\left(\frac{1}{2}\right)^{n-1}$.
  - Wkt., dass Charakteristik für zweites Klartext-Chiffrat-Paar den gleichen Wert annimmt wie für das erste, ...
  - (vereinfacht, bei der ersten Charakteristik ist die Wkt. sogar höchstens $\left(\frac{1}{2}\right)^{n}$)

ÜB2 – Aufgabe 3(d)

- Verwende die zwei Charakteristiken aus 3(b)
- Ereignis E_i: Angriff auf K_i erfolgreich

$$
\begin{aligned}
\Pr[E_4] &= \Pr\left[\begin{array}{l}\text{jedes falsche } K_4 \text{ erfüllt mind.}\\ \text{eine Charakteristik nicht}\end{array}\right] \\
&= \prod_{K_4 \text{ falsch}} 1 - 2^{-2n-2} \\
&= (\underbrace{1 - 2^{-2n-2}}_{\leq 1})^{2^{16}-1} \\
&\geq (1 \underbrace{-2^{-2n-2}}_{\geq -1})^{2^{16}} \\
&\overset{\text{Bernoulli}}{\geq} 1 - 2^{-2n-2} \cdot 2^{16} = 1 - 2^{14-2n}
\end{aligned}
$$

ÜB2 – Aufgabe 3(d)

- Ereignis E_i: Angriff auf K_i erfolgreich
- $\Pr[E_1] = \Pr[E_4]$

$$\begin{aligned} \Pr[\text{Angriff nicht erfolgreich}] &= \Pr[\neg E_1 \vee \neg E_4] \\ &\overset{\text{Union Bound}}{\leq} \Pr[\neg E_1] + \Pr[\neg E_4] \\ &\leq 2 \cdot (2^{14-2n}) \end{aligned}$$

- $\rightsquigarrow \Pr[\text{Angriff erfolgreich}] \geq 1 - 2^{15-2n}$
- Für z.B. $n = 12$ ist Erfolgswahrscheinlichkeit schon größer als 99%.

Socrative: Lineare Kryptoanalyse und Hashfunktionen

`https://b.socrative.com/login/student/`

Room: `SICHERHEIT`

- App um Quiz durchzuführen
- Zugang durch Browser oder App
- Als Quizteilnehmer kein Account notwendig.

Hashfunktionen

- „Fingerabdruck“ $H_k : \{0,1\}^* \to \{0,1\}^k$
- k ist Sicherheitsparameter
- **Kollisionsresistenz:** Für alle PPT-Algorithmen $\mathcal{A}$ ist

$$\Pr[(X, X') \leftarrow \mathcal{A}(1^k) : \; X \neq X' \wedge H_k(X) = H_k(X')]$$

vernachlässigbar.

Hashfunktionen: Beispiele

Sei $H : \{0,1\}^* \to \{0,1\}^k$ eine kollisionsresistente Hashfunktion.
Sind die folgenden Hashfunktionen ebenfalls kollisionsresistent?

- $H'(x) := H(f(x))$ (f injektiv)
- $H^*(x) := H(x) \oplus H(\overline{x})$ ($\overline{x}$ = bitweises Inverse von x)

Hashfunktionen: H'

$H'(x) := H(f(x))$ ist kollisionsresistent, denn:

- Angenommen es existiert PPT $\mathcal{A}$, der x, x' berechnet mit

$$H'(x) = H'(x') \text{ und } x \neq x'$$

  (mit nicht vernachlässigbarer Wkt.).
- Dann gilt

$$H'(x) = H(f(x)) = H(f(x')) = H'(x').$$

- Somit $y := f(x)$, $y' := f(x')$ Kollision für H.
- Außerdem: $x \neq x' \Rightarrow y \neq y'$.
- Konstruiere aus $\mathcal{A}$ Angreifer gegen H.
- Widerspruch zur Kollsisionsresistenz von H.

D.h. $\mathcal{A}$ kann nicht existieren und H' ist kollisionsresistent.

Hashfunktion: H^*

$H^*(x) := H(x) \oplus H(\overline{x})$ ist *nicht* kollisionsresistent, denn:

$$H^*(0) = H(0) \oplus H(1) = H(1) \oplus H(0) = H^*(1)$$

- Somit ist $0, 1$ eine (von vielen möglichen) Kollisionen für H^*...
- ... die effizient gefunden werden können.
- Generell gilt: $H^*(x) = H^*(\overline{x})$

Kollisionsresistenz: „Rezept“

Sei H kollisionsresistent, H' aus H konstruiert (wie eben).

Um Kollisionsresistenz zu zeigen:

- Konstruiere aus Kollision für H' eine Kollision für H.
- (Kollision muss effizient berechenbar sein!)

Um zu zeigen, dass H' nicht kollisionsresistent:

- Zeige, wie man effizient eine Kollision berechnen kann!

Kollisionsresistenz: „Rezept“

Zeige: H ist kollisionsresistent, weil Problem schwierig.
(Problem z.B. RSA, diskreter Logarithmus...)

Generelles Vorgehen

- Widerspruchsbeweis durch Reduktion
- Zeige: Kollisionen leicht berechenbar $\rightarrow$ Problem leicht lösbar

Kollisionsresistenz: „Rezept“

Auf Übungsblatt:

- $\mathbb{G}$ zyklische Gruppe, prime Ordnung, g Erzeuger
- DLog: Geg. g, g^x, berechne x (x zufällig gezogen)
- Annahme: DLog schwierig
- Vorgehen:
  - Konstruiere aus geg. g, g^x die Hashfunktion
  - Zeige: Kollision leicht berechenbar $\rightarrow$ x leicht berechenbar