Beweisbar sichere Verschlüsselung

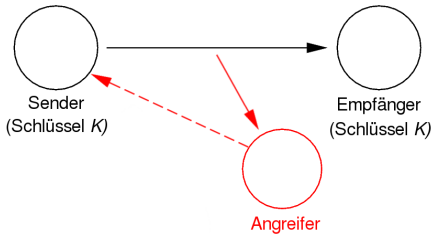
ITS-Wahlpflichtvorlesung

Dr. Bodo Möller
Ruhr-Universität Bochum
Horst-Görtz-Institut für IT-Sicherheit
Lehrstuhl für Kommunikationssicherheit
bmoeller@crypto.rub.de

$Id: bsv.ltx,v 1.76 2007/06/13 16:40:03 bm Exp $

Rückblick: CPA und CCA

- *CPA*-Angriffsmodelle (*Chosen Plaintext Attack*):

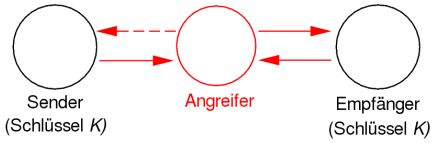


  Verschlüsselungorakel $E(\cdot)$

- Details für $E(\cdot)$ je nach Angriffsspiel:
  (z. B.) RoR, LoR; ggf. OT (= Einmalverschlüsselung)

Rückblick: CPA und CCA (Forts.)

- *CCA*-Angriffsmodelle (*Chosen Ciphertext Attack*):

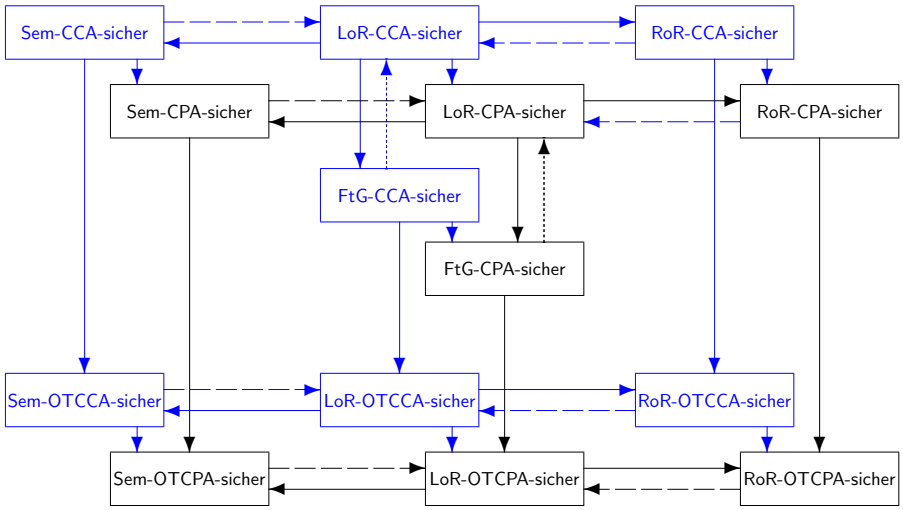


Verschlüsselungorakel $E(\cdot)$, *Entschlüsselungsorakel* $D(\cdot)$

- $E(\cdot)$ auch hier je nach Angriffsspiel (z. B.) RoR, LoR; ggf. OT
- $D(\cdot)$ fast uneingeschränkt:
  - echtes Entschlüsselungsorakel (kein "RoR" o. ä.!)
  - auch bei OT mehrfach verwendbar
  - als Anfragen an $D(\cdot)$ sind nur Antworten von $E(\cdot)$ nicht erlaubt
- Jeder CPA-Angriff ist auch ein (genauso erfolgreicher) CCA-Angriff!

Rückblick: CPA und CCA (Forts.)

- Wir wollen die neuen Sicherheitsbegriffe einordnen:



Rückblick: CPA und CCA (Forts.)

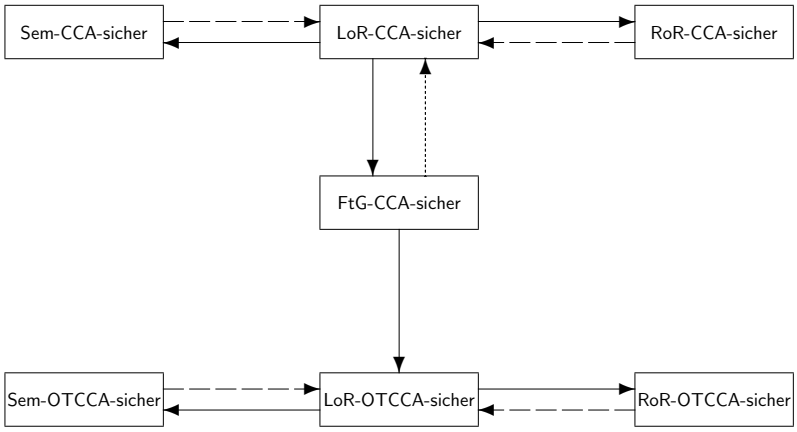
- *CCA*-Sicherheit ist eine *höhere Anforderung* als CPA-Sicherheit!
- Das hat Aufgabenblatt 5 gezeigt:
  CPA-Sicherheit impliziert nicht einmal OTCCA-Sicherheit

  Angriffsidee für CCA-Angriffe:
  Antworten von $E(\cdot)$ dürfen *nicht unverändert* an $D(\cdot)$ gegeben werden;
  aber *kleine Änderungen reichen aus*
  (z. B.: kippe das letzte Bit, hänge weitere Bits an),
  so kann man mit Hilfe von $D(\cdot)$ u. U. doch alles entschlüsseln
- Wie kann ein CCA-sicheres Verschlüsselungsschema aussehen ...? → Später!
- Ein *Entschlüsselungorakel* wie $D(\cdot)$ wirkt recht *artifiziell.*

  Entsprechung *in der Praxis* z. B.:
  Angreifer verfälscht Ciphertexte auf dem Weg vom Sender zum Empfänger,
  zieht Rückschlüsse aus der Reaktion des Empfängers
  (→ *partielle Information* über Plaintext)

Rückblick: CPA und CCA (Forts.)

- Davon noch anzusehen:



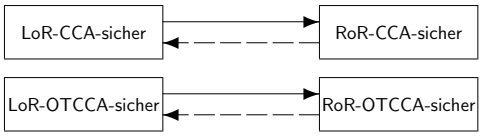
- ... mit Sem-CCA und Sem-OTCCA beschäftigen wir uns aber nicht!
  (Sicherheitsdefinition mit Simulator – vgl. Sem-OTCPA)
- Bleiben LoR-CCA, RoR-CCA; FtG-CCA; LoR-OTCCA, RoR-OTCCA

Vorschau

- *Zusammenhänge* zwischen CCA-Sicherheitsbegriffen
- *INT-CTXT* (Integrity of Ciphertexts)
- Nächste Woche: *Konstruktion* für CCA-sichere Verschlüsselung

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-(OT)CCA und RoR-(OT)CCA

- Jetzt geht es um:



- Das heißt (Kontraposition!):
  - Gegeben einen RoR-(OT)CCA-Angreifer, beschreibe einen *gleich erfolgreichen* LoR-(OT)CCA-Angreifer
  - Gegeben einen LoR-(OT)CCA-Angreifer, beschreibe einen *fast so erfolgreichen* RoR-(OT)CCA-Angreifer

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-(OT)CCA und RoR-(OT)CCA (Forts.)

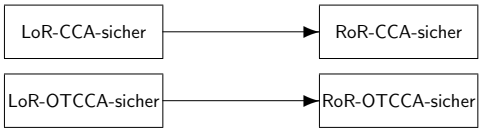
- Sei ein RoR-(OT)CCA-Angreifer $\boldsymbol{A}^{E(\cdot),D(\cdot)}$ gegeben.
  Wir werden daraus einen LoR-(OT)CCA-Angreifer $\boldsymbol{B}^{E'(\cdot,\cdot),D(\cdot)}$ konstruieren.
- Unterschied zwischen den Angriffsspielen:
  - $\boldsymbol{A}$ erwartet ein "Real-or-Random"-Verschlüsselungsorakel (ggf. "OT")
    mit Fällen $E_1(\cdot)$ ("real") und $E_0(\cdot)$ ("random")
  - $\boldsymbol{B}$ hat ein "Left-or-Right"-Verschlüsselungsorakel (ggf. "OT")
    mit Fällen $E'_1(\cdot,\cdot)$ ("left") und $E'_0(\cdot,\cdot)$ ("right")
- $\boldsymbol{B}$ lässt $\boldsymbol{A}$ ablaufen und ...
  - beantwortet Anfragen $E(m)$ (bei OT: *die* Anfrage)
    mit $E'(m, m_0)$, wobei $m_0 \in_\$ \{0,1\}^{|m|}$
  - reicht Anfragen $D(c)$ ans eigene Entschlüsselungsorakel durch
  - übernimmt das Ausgabebit $\widetilde{b}$ von $\boldsymbol{A}$
- Fall "left" für $\boldsymbol{B}$ ergibt genau Fall "real" für $\boldsymbol{A}$,
  Fall "right" für $\boldsymbol{B}$ ergibt genau Fall "random" für $\boldsymbol{A}$!

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-(OT)CCA und RoR-(OT)CCA (Forts.)

- Fall "left" für $\boldsymbol{B}$ ergibt genau Fall "real" für $\boldsymbol{A}$,
  Fall "right" für $\boldsymbol{B}$ ergibt genau Fall "random" für $\boldsymbol{A}$
- Also ist

$$\begin{aligned}\mathrm{Adv}^{\text{LoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} &= \Pr\left[\boldsymbol{B}^{E'_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] - \Pr\left[\boldsymbol{B}^{E'_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\ &= \Pr\left[\boldsymbol{A}^{E_1(\cdot),D(\cdot)} \Rightarrow 1\right] - \Pr\left[\boldsymbol{A}^{E_0(\cdot),D(\cdot)} \Rightarrow 1\right] \\ &= \mathrm{Adv}^{\text{RoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}}\end{aligned}$$

- Damit haben wir



Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-(OT)CCA und RoR-(OT)CCA (Forts.)

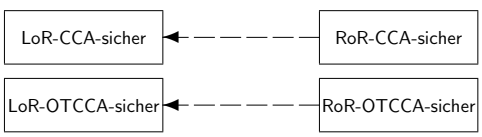
- Nun sei ein LoR-(OT)CCA-Angreifer $\boldsymbol{A}^{E(\cdot,\cdot),D(\cdot)}$ gegeben.
  Wir werden daraus einen RoR-(OT)CCA-Angreifer $\boldsymbol{B}^{E'(\cdot),D(\cdot)}$ konstruieren.
- Unterschied zwischen den Angriffsspielen:
  - $\boldsymbol{A}$ erwartet ein "Left-or-Right"-Verschlüsselungsorakel (ggf. "OT") mit Fällen $E_1(\cdot,\cdot)$ ("left") und $E_0(\cdot,\cdot)$ ("right")
  - $\boldsymbol{B}$ hat ein "Real-or-Random"-Verschlüsselungsorakel (ggf. "OT") mit Fällen $E'_1(\cdot)$ ("real) und $E'_0(\cdot)$ ("random")
- $\boldsymbol{B}$ wählt $x \in_\$ \{0,1\}$ und lässt dann $\boldsymbol{A}$ ablaufen und ...
  - beantwortet Anfragen $E(m_1,m_0)$ (bei OT: *die* Anfrage) mit $E'(m_x)$
  - reicht Anfragen $D(c)$ ans eigene Entschlüsselungsorakel durch
  - gibt selbst 1 aus, falls $\boldsymbol{A} \Rightarrow x$ (falls $\boldsymbol{A}$ also x erraten hat), 0 sonst
- Im "Real"-Fall von $\boldsymbol{B}$ ergibt sich das LoR-(OT)CCA-Angriffsspiel für $\boldsymbol{A}$: $x=1$ ist "left", $x=0$ ist "right"
- Im "Random"-Fall hat x keinen Einfluss darauf, was $\boldsymbol{A}$ erlebt

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-(OT)CCA und RoR-(OT)CCA (Forts.)

- Im "Real"-Fall von $\boldsymbol{B}$ ergibt sich das LoR-(OT)CCA-Angriffsspiel für $\boldsymbol{A}$: $x=1$ ist "left", $x=0$ ist "right"
- Im "Random"-Fall hat x keinen Einfluss darauf, was $\boldsymbol{A}$ erlebt
- Also (da allgemein $\Pr[\boldsymbol{A}^{E_1} \Rightarrow 1] - \Pr[\boldsymbol{A}^{E_0} \Rightarrow 1] = 2 \cdot \Pr_{b\in\{0,1\}}[\boldsymbol{A}^{E_b} \Rightarrow b] - 1$)

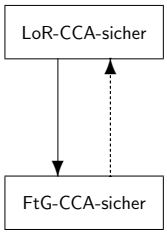
$$\begin{aligned}\mathrm{Adv}^{\text{RoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} &= \Pr\left[\boldsymbol{B}^{E'_1(\cdot),D(\cdot)} \Rightarrow 1\right] - \Pr\left[\boldsymbol{B}^{E'_0(\cdot),D(\cdot)} \Rightarrow 1\right] \\ &= \Pr\left[\boldsymbol{B}^{E'_1(\cdot),D(\cdot)} \Rightarrow 1\right] - \frac{1}{2} \\ &= \Pr_{x\in_\$\{0,1\}}\left[\boldsymbol{A}^{E_x(\cdot,\cdot),D(\cdot)} \Rightarrow x\right] - \frac{1}{2} \quad = \quad \frac{1}{2}\cdot \mathrm{Adv}^{\text{LoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}}\end{aligned}$$

- Damit haben wir



Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA

- Jetzt geht es um:



- Das heißt (Kontraposition!):
  - Gegeben einen FtG-CCA-Angreifer, beschreibe einen *gleich erfolgreichen* LoR-CCA-Angreifer
  - Gegeben einen LoR-CCA-Angreifer, beschreibe einen *(u. U. weniger erfolgreichen)* FtG-CCA-Angreifer

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA (Forts.)

- Sei ein FtG-CCA-Angreifer $\boldsymbol{A}$ gegeben. $\boldsymbol{A}$ erwartet drei Orakel:
  - „Echtes“ Verschlüsselungsorakel $E(\cdot)$
  - Left-or-Right-Verschlüsselungsorakel mit Fällen $E_1(\cdot,\cdot)$ (“left”) und $E_0(\cdot,\cdot)$ (“right”)
  - Entschlüsselungsorakel $D(\cdot)$

  $\boldsymbol{A}$ darf $E(\cdot)$ und $D(\cdot)$ mehrfach befragen, $E_b(\cdot,\cdot)$ jedoch nur einmal.
  Wie in einem LoR-Angriffsspiel geht es für $\boldsymbol{A}$ darum, $E_1(\cdot,\cdot)$ und $E_0(\cdot,\cdot)$ zu unterscheiden.
  Vorteil:

$$\mathrm{Adv}^{\text{FtG-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}} = \Pr\left[\boldsymbol{A}^{E(\cdot),E_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] - \Pr\left[\boldsymbol{A}^{E(\cdot),E_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right]$$

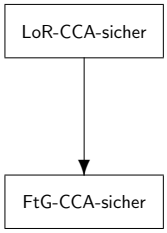
- Wir werden aus $\boldsymbol{A}$ einen LoR-CCA-Angreifer $\boldsymbol{B}^{E'(\cdot,\cdot),D(\cdot)}$ konstruieren.

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA (Forts.)

- Sei ein FtG-CCA-Angreifer $\boldsymbol{A}$ gegeben.
  Wir werden aus $\boldsymbol{A}$ einen LoR-CCA-Angreifer $\boldsymbol{B}^{E'(\cdot,\cdot),D(\cdot)}$ konstruieren.
- $\boldsymbol{B}$ lässt $\boldsymbol{A}$ ablaufen und ...
  - beantwortet Anfragen $E(m)$ mit $E'(m,m)$
  - beantwortet Anfragen $E(m_1,m_0)$ mit $E'(m_1,m_0)$
  - reicht Anfragen $D(c)$ ans eigene Entschlüsselungsorakel durch
  - übernimmt das Ausgabebit $\widetilde{b}$ von $\boldsymbol{A}$
- Damit läuft $\boldsymbol{A}$ genau im FtG-CCA-Angriffsspiel ab:
  "left" oder "right" so wie bei $\boldsymbol{B}$
- $$\begin{aligned}\mathrm{Adv}^{\text{LoR-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} &= \Pr\left[\boldsymbol{B}^{E'_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] - \Pr\left[\boldsymbol{B}^{E'_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\ &= \Pr\left[\boldsymbol{A}^{E(\cdot),E_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] - \Pr\left[\boldsymbol{A}^{E(\cdot),E_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\ &= \mathrm{Adv}^{\text{FtG-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}}\end{aligned}$$

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA (Forts.)

- Wir haben $\boldsymbol{B}$ aus $\boldsymbol{A}$ konstruiert mit $\mathrm{Adv}^{\text{LoR-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} = \mathrm{Adv}^{\text{FtG-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}}$
- Damit haben wir gezeigt:



Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA (Forts.)

- Sei nun ein LoR-CCA-Angreifer $\boldsymbol{A}^{E(\cdot,\cdot),D(\cdot)}$ gegeben.
  Wir werden daraus einen FtG-CCA-Angreifer $\boldsymbol{B}^{E'(\cdot),E'(\cdot,\cdot),D(\cdot)}$ konstruieren.
- Hier müssen wir einen größeren Unterschied beim Vorteil hinnehmen (vgl. früher FtG-CPA und LoR-CPA!)
- Annahme: $\boldsymbol{A}$ benutze sein Left-or-Right-Orakel höchstens q mal
- Konstruiere aus $\boldsymbol{A}$ zunächst FtG-CCA-Angreifer $\boldsymbol{B}_i$ ($i \in \{1,\ldots,q\}$):
  - Die ersten $i-1$ Anfragen von $\boldsymbol{A}$ an $E(\cdot,\cdot)$ werden mit Hilfe von $E'(\cdot)$ mit *Links*verschlüsselungen beantwortet
  - Die i-te Anfrage von $\boldsymbol{A}$ an $E(\cdot,\cdot)$ wird an $E'(\cdot,\cdot)$ weitergereicht (Left-or-Right)
  - Die verbleibenden Anfragen von $\boldsymbol{A}$ an $E(\cdot,\cdot)$ werden mit Hilfe von $E'(\cdot)$ mit *Rechts*verschlüsselungen beantwortet

  Anfragen an $D(\cdot)$ werden direkt weitergereicht,
  das Ausgabebit wird von $\boldsymbol{A}$ übernommen.

Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA (Forts.)

- Beim "Right"-Fall von $\boldsymbol{B}_1$ sieht $\boldsymbol{A}$ durchgängig Rechtsverschlüsselung;
  beim "Left"-Fall von $\boldsymbol{B}_q$ durchgängig Linksverschlüsselung
- Alles andere sind „Hybride“:
  $\boldsymbol{A}$ hat anfangs Linksverschlüsselung, später Rechtsverschlüsselung
- Der "Left"-Fall von $\boldsymbol{B}_i$ ist stets ($1 \leq i < q$) das gleiche wie der "Right"-Fall von $\boldsymbol{B}_{i+1}$ (nämlich i-mal Links-, dann Rechtsverschlüsselung)!

  Also $\Pr\left[\boldsymbol{B}_i^{E'(\cdot),E'_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] = \Pr\left[\boldsymbol{B}_{i+1}^{E'(\cdot),E'_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right]$
- Setze $\boldsymbol{B}_1,\ldots,\boldsymbol{B}_q$ zusammen zu einem einzigen FtG-Angreifer:
  Wähle ganz am Anfang $i \in_\$ \{1,\ldots,q\}$, verwende dann $\boldsymbol{B}_i$

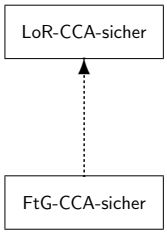
Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA (Forts.)

- Wähle ganz am Anfang $i \in_\$ \{1, \ldots, q\}$, verwende dann $\boldsymbol{B}_i$
- Dann ist (wegen $\Pr\left[\boldsymbol{B}_i^{E'(\cdot),E'_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] = \Pr\left[\boldsymbol{B}_{i+1}^{E'(\cdot),E'_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right]$)

$$\begin{aligned}
\mathrm{Adv}^{\text{FtG-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} &= \Pr\left[\boldsymbol{B}^{E'(\cdot),E'_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] - \Pr\left[\boldsymbol{B}^{E'(\cdot),E'_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\
&= \frac{1}{q} \cdot \sum_{1 \le i \le q} \Pr\left[\boldsymbol{B}_i^{E'(\cdot),E'_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\
&\quad - \frac{1}{q} \cdot \sum_{1 \le i \le q} \Pr\left[\boldsymbol{B}_i^{E'(\cdot),E'_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\
&= \frac{1}{q} \cdot \Pr\left[\boldsymbol{B}_q^{E'(\cdot),E'_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] - \frac{1}{q} \cdot \Pr\left[\boldsymbol{B}_1^{E'(\cdot),E'_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\
&= \frac{1}{q} \cdot \Pr\left[\boldsymbol{A}_q^{E_1(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] - \frac{1}{q} \cdot \Pr\left[\boldsymbol{A}_1^{E_0(\cdot,\cdot),D(\cdot)} \Rightarrow 1\right] \\
&= \frac{1}{q} \cdot \mathrm{Adv}^{\text{LoR-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}}
\end{aligned}$$

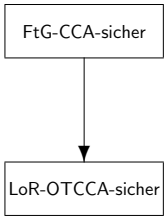
Zusammenhänge zwischen CCA-Sicherheitsbegriffen: LoR-CCA und FtG-CCA (Forts.)

- $\mathrm{Adv}^{\text{FtG-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} = \frac{1}{q} \cdot \mathrm{Adv}^{\text{LoR-CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}}$
- Damit haben wir



Zusammenhänge zwischen CCA-Sicherheitsbegriffen: FtG-CCA und LoR-OTCCA

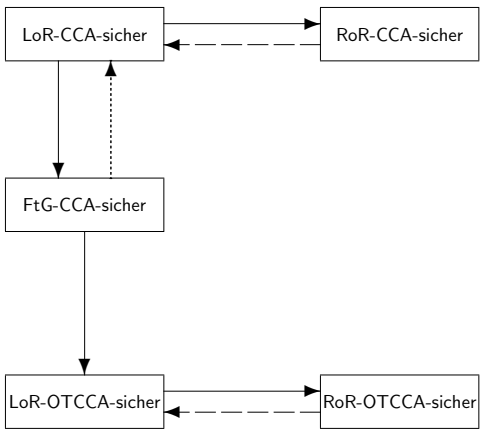
- Jetzt geht es um:



- Das heißt (Kontraposition!):
  - Gegeben einen LoR-OTCCA-Angreifer, beschreibe einen *gleich erfolgreichen* FtG-CCA-Angreifer
- Der LoR-OTCCA-Angreifer *ist* auch ein FtG-CCA-Angreifer (nur eine Benuztung des LoR-Verschlüsselungsorakels wegen "OT")
- Fertig!

Zusammenhänge zwischen CCA-Sicherheitsbegriffen

- Alles in allem haben wir gesehen:



- Aber wie erreichen wir CCA-sichere Verschlüsselung eigentlich ...?

INT-CTXT: Integrity of Ciphertexts

- Aufgabenblatt 5 hat gezeigt:
  Verschlüsselung mit Counter Mode oder mit CBC ist nicht CCA-sicher,
  CCA-Sicherheit ist also eine *höhere Anforderung* als *CPA-Sicherheit*
- Antworten vom jeweiligen Verschlüsselungsorakel dürfen zwar nicht *unverändert* ans Entschlüsselungsorakel $D(\cdot)$ gegeben werden;
  aber mit kleinen Änderungen am Ciphertext kann der Angreifer $D(\cdot)$ indirekt u. U. doch für alles verwenden
- Wir brauchen einen Ansatz, um solche Angriffe zu verhindern
- Eine Idee: Versuche, ein Verschlüsselungsschema so zu konstruieren, dass ein Entschlüsselungsorakel überhaupt nichts beitragen kann
- Formalisierung dieser Eigenschaft als *Integrity of Ciphertexts* ...

INT-CTXT: Integrity of Ciphertexts (Forts.)

- *INT-CTXT-Angriffsspiel* auf Verschlüsselungsschema $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ für einen Angreifer $\boldsymbol{A}$:
  - $K \xleftarrow{\$} \mathcal{K}$ (nicht sichtbar für $\boldsymbol{A}$)
  - $\boldsymbol{A}$ bekommt Zugriff auf Orakel $E(\cdot)$ und $D(\cdot)$:
    $E(m)$ liefert $\mathcal{E}_K(m)$ (*Verschlüsselungsorakel*),
    $D(c)$ liefert $\mathcal{D}_K(c)$ (*Entschlüsselungsorakel*)
  - Anfrage $D(c)$ ist nicht erlaubt, wenn c eine der früheren Antworten von $E(\cdot)$ war
  - $\boldsymbol{A}$ gewinnt das Spiel, wenn $D(c)$ jemals etwas anderes als $\perp$ antwortet
- *INT-CTXT-Vorteil:*

$$\mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}} = \Pr\left[\boldsymbol{A}^{E(\cdot),D(\cdot)} \text{ gewinnt}\right]$$

- $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ ist *INT-CTXT-sicher* (bietet Ciphertext-Integrität), wenn dieser Vorteil für jeden denkbaren Angreifer verschwindend gering bleibt

INT-CTXT: Integrity of Ciphertexts (Forts.)

Zusammenhang mit CCA-Sicherheit und CPA-Sicherheit

- Angenommen, wir haben ein Verschlüsselungsschema $(\mathcal{K}, \mathcal{E}, \mathcal{D})$, das sowohl LoR-(OT)CPA-sicher ist als auch INT-CTXT-sicher
- Wir wollen zeigen: $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ muss dann LoR-(OT)CCA-sicher sein
- Sei $\boldsymbol{A}$ also ein LoR-(OT)CCA-Angreifer
- Wir konstruieren hier gleich zwei Angreifer aus $\boldsymbol{A}$:
  - einen LoR-(OT)CPA-Angreifer $\boldsymbol{B}$,
  - einen INT-CTXT-Angreifer $\boldsymbol{C}$

INT-CTXT: Integrity of Ciphertexts (Forts.)

- Sei $\boldsymbol{A}$ ein LoR-(OT)CCA-Angreifer. Wir konstruieren aus $\boldsymbol{A}$
  - einen LoR-(OT)CPA-Angreifer $\boldsymbol{B}$,
  - einen INT-CTXT-Angreifer $\boldsymbol{C}$
- Angreifer $\boldsymbol{B}$ lässt $\boldsymbol{A}$ laufen und geht genauso vor;
  Ausnahme: wo $\boldsymbol{A}$ das Entschlüsselungsorakel befragt (das $\boldsymbol{B}$ fehlt), verwendet $\boldsymbol{B}$ immer $\perp$ als Antwort
- Angreifer $\boldsymbol{C}$ wählt $x \in_\$ \{0,1\}$ und lässt dann $\boldsymbol{A}$ ablaufen und ...
  - beantwortet Anfragen $E(m_1, m_0)$ (bei OT: *die* Anfrage) mit $E(m_x)$
  - reicht Anfragen $D(c)$ ans eigene Entschlüsselungsorakel durch

  Die Ausgabe von $\boldsymbol{A}$ wird ignoriert ($\boldsymbol{C}$ braucht kein Ergebnis!)
- Wir suchen jetzt einen Zusammenhang zwischen $\mathrm{Adv}^{\text{LoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}}$, $\mathrm{Adv}^{\text{LoR-(OT)CPA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}}$ und $\mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{C}}$

INT-CTXT: Integrity of Ciphertexts (Forts.)

- Wegen $\mathrm{Adv}^{\text{LoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}} = 2 \cdot \Pr_{b\in_\$\{0,1\}}\left[\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right] - 1$
  interessieren wir uns für $\Pr_{b\in_\$\{0,1\}}\left[\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right]$
- Sei F das Ereignis, dass $\boldsymbol{A}$ einen neuen Ciphertext „fälscht“,
  also jemals von $D(\cdot)$ eine Antwort bekommt
- Dann ist
  $\Pr_{b\in_\$\{0,1\}}\left[\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right] =$
  $\Pr_{b\in_\$\{0,1\}}\left[F \wedge \left(\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right)\right] + \Pr_{b\in_\$\{0,1\}}\left[\overline{F} \wedge \left(\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right)\right]$
- Betrachten wir den INT-CTXT-Angreifer $\boldsymbol{C}$:
  - wählt $x \in_\$ \{0,1\}$,
  - beantwortet Anfragen $E(m_1,m_0)$ von $\boldsymbol{A}$ mit $E(m_x)$,
  - verwendet Anfragen $D(c)$ direkt.

  Also $\mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{C}} = \Pr[F]$
- Somit $\Pr_b\left[\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right] \le \mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{C}} + \Pr_b\left[\overline{F} \wedge \left(\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right)\right]$
- Betrachten wir den LoR-(OT)CPA-Angreifer $\boldsymbol{B}$: ...

INT-CTXT: Integrity of Ciphertexts (Forts.)

- $\Pr_b\left[\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right] \le \mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{C}} + \Pr_b\left[\overline{F} \wedge \left(\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right)\right]$
- Betrachten wir den LoR-(OT)CPA-Angreifer $\boldsymbol{B}$:
  $\boldsymbol{B}$ stimmt mit $\boldsymbol{A}$ überein für das Ereignis $\overline{F}$, also
  $$\Pr_{b\in_\$\{0,1\}}\left[\overline{F} \wedge \left(\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right)\right] \le \Pr_{b\in_\$\{0,1\}}\left[\boldsymbol{B}^{E_b(\cdot,\cdot)} \Rightarrow b\right]$$
- Wegen
  $$\Pr_{b\in_\$\{0,1\}}\left[\boldsymbol{B}^{E_b(\cdot,\cdot)} \Rightarrow b\right] = \frac{1}{2} \cdot \mathrm{Adv}^{\text{LoR-(OT)CPA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} + \frac{1}{2}$$
  folgt jetzt
  $$\Pr_{b\in_\$\{0,1\}}\left[\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right] \le \mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{C}} + \frac{1}{2} \cdot \mathrm{Adv}^{\text{LoR-(OT)CPA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}} + \frac{1}{2}$$
- ... und wegen $\mathrm{Adv}^{\text{LoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}} = 2 \cdot \Pr_{b\in_\$\{0,1\}}\left[\boldsymbol{A}^{E_b(\cdot,\cdot),D(\cdot)} \Rightarrow b\right] - 1$ also
  $$\mathrm{Adv}^{\text{LoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}} \le 2 \cdot \mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{C}} + \mathrm{Adv}^{\text{LoR-(OT)CPA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}},$$
  die gewünschte Ungleichung von Vorteilen!

INT-CTXT: Integrity of Ciphertexts (Forts.)

- Die Ungleichung

$$\mathrm{Adv}^{\text{LoR-(OT)CCA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{A}} \leq 2 \cdot \mathrm{Adv}^{\text{INT-CTXT}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{C}} + \mathrm{Adv}^{\text{LoR-(OT)CPA}}_{(\mathcal{K},\mathcal{E},\mathcal{D}),\boldsymbol{B}}$$

  sagt uns (weil $\boldsymbol{B}$ und $\boldsymbol{C}$ direkt von $\boldsymbol{A}$ abgeleitet sind mit im wesentlichen gleichen Ressourcenbedarf):

  INT-CTXT-Sicherheit und LoR-(OT)CPA-Sicherheit zusammen impliziert LoR-(OT)CCA-Sicherheit!

- Aber INT-CTXT ist auch aus sich heraus relevant:
  Empfänger kann sehr sicher sein, dass jeder entschlüsselbare Ciphertext vom Sender selbst verschlüsselt wurde