Sicherheitsmodelle

4

In Kap. 3 wurden eine Reihe von historischen Verschlüsselungsverfahren beschrieben. Es stellte sich heraus, dass sie alle affin linear und daher unsicher sind. Es fragt sich also, ob es mathematisch beweisbar sichere Verschlüsselungsverfahren gibt. Um diese Frage beantworten zu können, führen wir in diesem Kapitel mathematische Modelle der Sicherheit von Verschlüsselungsverfahren ein. Anschließend diskutieren wir, ob es Kryptosysteme gibt, die gemäß dieser Modelle beweisbar sicher sind. Das erste solche Sicherheitsmodell und ein in diesem Modell sicheres Verschlüsselungsverfahren wurde 1949 von Claude Shannon [68] vorgestellt. Wir beginnen mit der Beschreibung dieses Modells. Es stellt sich dabei heraus, dass der Sicherheitsbegriff von Shannon zu restriktiv ist. Er wurde deshalb in den 1980er Jahren weiterentwickelt. Grundlegend dafür sind Arbeiten von Shafi Goldwasser und Silvio Micali (siehe [32]). Dieses Kapitel diskutiert auch das weiterentwickelte Modell.

4.1 Perfekte Geheimhaltung

In diesem Abschnitt erläutern wir das Modell der perfekten Geheimhaltung von Claude Shannon.

Im Modell von Shannon benutzt Alice ein Kryptosystem **E** mit endlichem Klartextraum **P**, endlichem Schlüsseltextraum **C** und endlichem Schlüsselraum **K**. Der Schlüsselerzeugungsalgorithmus **KeyGen** wählt Schlüssel aus dem Schlüsselraum. Jeder Schlüssel wird aber nur einmal zum Verschlüsseln verwendet. Danach wird ein neuer Schlüssel gewählt. Verschlüsselungsalgorithmus und Entschlüsselungsalgorithmus sind deterministisch. Die entsprechenden Verschlüsselungs- und Entschlüsselungsfunktionen werden mit **Enc** und **Dec** bezeichnet.

Das Modell erlaubt Angreifern nur Ciphertext-Only-Angriffe. Komplexere Angriffe, also Chosen-Plaintext-Angriffe oder Chosen-Ciphertext-Angriffe, sind nämlich nicht möglich, weil jeder Schlüssel nur einmal verwendet werden kann. Darum ist es auch

J. Buchmann, *Einführung in die Kryptographie*, Springer-Lehrbuch,
DOI 10.1007/978-3-642-39775-2_4

kein Problem, dass der Verschlüsslungsalgorithmus deterministisch ist, obwohl in Abschn. 3.4.2 erklärt wurde, dass deterministische Verschlüsselung immer Chosen-Plaintext-Angriffe möglich macht. Solche Angriffe setzen eben voraus, dass Schlüssel mehrfach verwendet werden.

Angreifer haben gewisse Kontextinformationen und wissen, dass nicht alle Klartexte gleich wahrscheinlich sind. Wenn die Kommunikationspartner zum Beispiel Deutsche sind, sind englische Klartexte eher unwahrscheinlich. Sind Alice und Bob von Beruf Lehrer, dann ist die Wahrscheinlichkeit dafür, dass in einer Nachricht von Alice an Bob das Wort „Schüler“ vorkommt, groß. Diese Kontextinformationen werden so modelliert, dass die Klartexte gemäß einer Wahrscheinlichkeitsverteilung $\mathrm{Pr}_{\mathbf{P}}$ auftreten. Sie ist möglichen Angreifern bekannt. Für einen Klartext P ist $\mathrm{Pr}_{\mathbf{P}}(P)$ die Wahrscheinlichkeit dafür, dass der Klartext P ausgewählt und verschlüsselt wird. Die Kenntnis von $\mathrm{Pr}_{\mathbf{P}}$ haben Angreifer a priori. Sie brauchen dafür keine Schlüsseltexte zu kennen.

Der Schlüsselerzeugungsalgorithmus wählt Schlüssel gemäß einer Wahrscheinlichkeitsverteilung $\mathrm{Pr}_{\mathbf{K}}$. Aus $\mathrm{Pr}_{\mathbf{P}}$ und $\mathrm{Pr}_{\mathbf{K}}$ erhält man eine Wahrscheinlichkeitsverteilung Pr auf $\mathbf{P} \times \mathbf{K}$. Für einen Klartext P und einen Schlüssel K ist $\mathrm{Pr}(P, K)$ die Wahrscheinlichkeit dafür, dass Alice den Klartext P wählt und mit dem Schlüssel K verschlüsselt. Es gilt

$$\mathrm{Pr}(P, K) = \mathrm{Pr}_{\mathbf{P}}(P)\,\mathrm{Pr}_{\mathbf{K}}(K) \tag{4.1}$$

und dadurch ist Pr festgelegt. Ab jetzt betrachten wir nur noch diese Wahrscheinlichkeitsverteilung. Sei $P \in \mathbf{P}$. Zur Vereinfachung bezeichnen wir mit P auch das Ereignis $\{(P, K) : K \in \mathbf{K}\}$. Tritt es auf, wird der Klartext P verschlüsselt. Mit dieser Schreibweise gilt wegen (4.1)

$$\mathrm{Pr}(P) = \sum_{K \in \mathbf{K}} \mathrm{Pr}(P, K) = \mathrm{Pr}_{\mathbf{P}}(P) \sum_{K \in \mathbf{K}} \mathrm{Pr}_{\mathbf{K}}(K) = \mathrm{Pr}_{\mathbf{P}}(P). \tag{4.2}$$

Dies rechtfertigt die Doppelbedeutung von P. Entsprechend bezeichnen wir mit $K \in \mathbf{K}$ auch das Ereignis, dass der Schlüssel K ausgewählt wird, also das Ereignis $\{(P, K): P \in \mathbf{P}\}$. Wie in (4.2) zeigt man

$$\mathrm{Pr}(K) = \mathrm{Pr}_{\mathbf{K}}(K). \tag{4.3}$$

Nach (4.1) sind die Ereignisse P und K unabhängig. Für $C \in \mathbf{K}$ bezeichne C das Ereignis, dass das Ergebnis der Verschlüsselung von P mit Schlüssel K der Schlüsseltext C ist, also das Ereignis $\{(P, K) : \mathbf{Enc}(K, P) = C\}$.

Jetzt erklären wir, wie Shannon die Sicherheit des Verschlüsselungsverfahrens $\mathbf{E}$ modelliert. Ein Angreifer kennt die Wahrscheinlichkeitsverteilung $\mathrm{Pr}_{\mathbf{P}}$ auf den Klartexten und hat damit eine Basisinformation, die ihm das Entschlüsseln erleichern kann. Jetzt sieht er einen Schlüsseltext C. Wann lernt er etwas aus C? Wenn das Auftreten von C die Wahrscheinlichkeitsverteilung auf den Klartexten verändert, wenn also manche Klartexte wahrscheinlicher als vorher sind und andere weniger wahrscheinlich, sobald der Schlüsseltext C aufgetreten ist. Mit anderen Worten: der Angreifer lernt nichts aus C, wenn sich

die Wahrscheinlichkeitsverteilung auf den Klartexten nach Auftreten von C nicht ändert. Dies rechtfertigt die folgende Definition.

Definition 4.1 Das Kryptosystem **E** heißt *perfekt geheim*, wenn die Ereignisse, dass ein bestimmter Schlüsseltext auftritt und dass ein bestimmter Klartext vorliegt, unabhängig sind, wenn also $\Pr(P|C) = \Pr(P)$ für alle Klartexte P und alle Schlüsseltexte C gilt.

Beispiel 4.1 Sei $\mathbf{P} = \mathbb{Z}_2$, $\Pr(0) = 1/4$, $\Pr(1) = 3/4$. Weiter sei $\mathbf{K} = \{A, B\}$, $\Pr(A) = 1/4$, $\Pr(B) = 3/4$. Schließlich sei $\mathbf{C} = \{a, b\}$. Dann ist die Wahrscheinlichkeit dafür, dass das Zeichen 1 auftritt und mit dem Schlüssel B verschlüsselt wird, $\Pr(1)\Pr(B) = 3/4 \cdot 3/4 = 9/16$. Die Verschlüsselungsfunktion **Enc** sei folgendermaßen definiert:

$$\mathbf{Enc}(A, 0) = a, \mathbf{Enc}(A, 1) = b, \mathbf{Enc}(B, 0) = b, \mathbf{Enc}(B, 1) = a.$$

Die Wahrscheinlichkeit dafür, dass der Schlüsseltext a auftritt, ist $\Pr(a) = \Pr(0, A) + \Pr(1, B) = 1/16 + 9/16 = 5/8$. Die Wahrscheinlichkeit dafür, dass der Schlüsseltext b auftritt, ist $\Pr(b) = \Pr(1, A) + \Pr(0, B) = 3/16 + 3/16 = 3/8$.

Wir berechnen nun die bedingten Wahrscheinlichkeiten $\Pr(P|C)$ für alle Klartexte P und alle Schlüsseltexte C. Es ist $\Pr(0|a) = 1/10$, $\Pr(1|a) = 9/10$, $\Pr(0|b) = 1/2$, $\Pr(1|b) = 1/2$. Diese Ergebnisse zeigen, dass das beschriebene Kryptosystem nicht perfekt geheim ist. Wenn Oskar zum Beispiel den Schlüsseltext a beobachtet, kann er ziemlich sicher sein, dass der zugehörige Klartext 1 war.

Wir stellen noch eine äquivalente Definition von perfekt sicherer Verschlüsselung vor, die bei der Einführung der Sicherheitsmodelle in den folgenden Abschnitten nützlich sein wird.

Theorem 4.1 *Das Kryptosystem* **E** *ist genau dann perfekt geheim, wenn für alle Paare* (P_0, P_1) *von Klartexten und für alle Schlüsseltexte* C *gilt:* $\Pr(C|P_0) = \Pr(C|P_1)$, *wenn also die Wahrscheinlichkeit dafür, dass beim Verschlüsseln ein Chiffretext* C *ensteht unabhängig vom verschlüsselten Klartext ist.*

Beweis Übung 4.3. □

Perfekt geheim heißt also, dass die Verteilungen auf den Schlüsseltexten bei Verschlüsselung zweier unterschiedlicher Klartexte identisch ist. Diese Charakterisierung ermöglicht es, zu zeigen, dass Verschlüsselungsverfahren, die durch Verwendung von Blockchiffren im ECB-Mode entstehen, keine perfekte Geheimhaltung bieten. Dies wird im nächsten Beispiel dargestellt.

Beispiel 4.2 Wir verwenden eine Blockchiffre über dem Alphabet $\mathbb{Z}_2$ mit Blocklänge n im ECB-Mode. Zur Vereinfachung der Beschreibung legen wir außerdem fest, dass der

ECB-Mode nur Klartexte verschlüsselt, die aus 2 Blöcken bestehen. Der für diesen Spezialfall gegebene Beweis der Unsicherheit kann leicht verallgemeinert werden. Klartext- und Chiffretextraum sind also $\mathbb{Z}_2^{2n}$. Die Wahrscheinlichkeitsverteilung auf den Klartexten und Schlüsseln sei die Gleichverteilung.

Sei $P_0 = 0^{2n}$, sei C_0 der entsprechende Chiffretext und sei $P_1 = 0^n 1^n$. Dann ist $C_0 = CC$ mit $C \in \mathbb{Z}_2^n$. Nun gilt $\Pr(C_0|P_0) > 0$ aber $\Pr(C_0|P_1) = 0$, weil zwei verschiedene Blöcke nie zu zwei gleichen Blöcken verschlüsselt werden. Also ist das Kryptosystem gemäß Theorem 4.1 nicht perfekt geheim.

Der berühmte Satz von Shannon, den wir jetzt beweisen, charakterisiert perfekt geheime Verschlüsselungsverfahren.

Theorem 4.2 *Sei* $|\mathbf{P}| = |\mathbf{K}| = |\mathbf{C}| < \infty$ *und sei* $\Pr(P) > 0$ *für jeden Klartext* P. *Das Kryptosystem* $\mathbf{E}$ *ist genau dann perfekt geheim, wenn die Wahrscheinlichkeitsverteilung auf dem Schlüsselraum die Gleichverteilung ist und wenn es für jeden Klartext* P *und jeden Schlüsseltext* C *genau einen Schlüssel* K *gibt mit* $\mathbf{Enc}(K, P) = C$.

Beweis Angenommen, das Verschlüsselungssystem ist perfekt geheim. Sei P ein Klartext. Wenn es einen Schlüsseltext C gibt, für den es keinen Schlüssel K gibt mit $\mathbf{Enc}(K, P) = C$, dann ist $0 < \Pr(P) \neq \Pr(P|C) = 0$. Aber dies widerspricht der perfekten Geheimhaltung. Für jeden Schlüsseltext C gibt es also einen Schlüssel K mit $\mathbf{Enc}(K, P) = C$. Da aber die Anzahl der Schlüssel gleich der Anzahl der Schlüsseltexte ist, gibt es für jeden Schlüsseltext C genau einen Schlüssel K mit $\mathbf{Enc}(K, P) = C$. Dies beweist die zweite Behauptung.

Um die erste Behauptung zu beweisen, fixiere einen Schlüsseltext C. Für einen Klartext P sei $K(P)$ der eindeutig bestimmte Schlüssel mit $\mathbf{Enc}(K(P), P) = C$. Weil es genauso viele Klartexte wie Schlüssel gibt, ist

$$\mathbf{K} = \{K(P) : P \in \mathbf{P}\} \tag{4.4}$$

Wir zeigen, dass für alle $P \in \mathbf{P}$ die Wahrscheinlichkeit für $K(P)$ gleich der Wahrscheinlichkeit für C ist. Dann ist die Wahrscheinlichkeit für $K(P)$ unabhängig von P. Da aber nach (4.4) jeder Schlüssel K mit einem $K(P)$, $P \in \mathbf{P}$ übereinstimmt, sind alle Schlüssel gleich wahrscheinlich.

Sei $P \in \mathbf{P}$. Wir zeigen $\Pr(K(P)) = \Pr(C)$. Nach Theorem 1.8 gilt für jeden Klartext P

$$\Pr(P|C) = \frac{\Pr(C|P)\Pr(P)}{\Pr(C)} = \frac{\Pr(K(P))\Pr(P)}{\Pr(C)}. \tag{4.5}$$

Weil das Verschlüsselungssystem perfekt geheim ist, gilt $\Pr(P|C) = \Pr(P)$. Aus (4.5) folgt daher $\Pr(K(P)) = \Pr(C)$, und dies ist unabhängig von P.

Wir beweisen die Umkehrung. Angenommen, die Wahrscheinlichkeitsverteilung auf dem Schlüsselraum ist die Gleichverteilung und für jeden Klartext p und jeden Schlüsseltext C gibt es genau einen Schlüssel $K = K(P, C)$ mit $\mathbf{Enc}(K, P) = C$. Dann folgt

$$\Pr(P|C) = \frac{\Pr(P)\Pr(C|P)}{\Pr(C)} = \frac{\Pr(P)\Pr(K(P,C))}{\sum_{Q\in\mathbf{P}}\Pr(Q)\Pr(K(Q,C))}. \tag{4.6}$$

Nun ist $\Pr(K(P, C)) = 1/|\mathbf{K}|$, weil alle Schlüssel gleich wahrscheinlich sind. Außerdem ist

$$\sum_{Q\in\mathbf{P}}\Pr(Q)\Pr(K(Q,C)) = \frac{\sum_{Q\in\mathbf{P}}\Pr(Q)}{|\mathbf{K}|} = \frac{1}{|\mathbf{K}|}.$$

Setzt man dies in (4.6) ein, so folgt $\Pr(P|C) = \Pr(P)$, wie behauptet. □

Beispiel 4.3 Aus Theorem 4.2 folgt, dass das Kryptosystem aus Beispiel 4.1 perfekt geheim wird, wenn man $\Pr(A) = \Pr(B) = 1/2$ setzt.

4.2 Das Vernam-One-Time-Pad

Das bekannteste Kryptosystem, dessen perfekte Geheimhaltung wir mit Theorem 4.2 beweisen können, ist das *Vernam-One-Time-Pad (OTP)*. Sei n eine natürliche Zahl. Das Vernam-One-Time-Pad verschlüsselt Bitstrings der Länge n. Schlüsselraum, Klartextraum und Chiffretextraum sind $\mathbf{K} = \mathbf{P} = \mathbf{C} = \mathbb{Z}_2^n$. Der Schlüsselerzeugungsalgorithmus wählt zufällig und gleichverteilt einen Schlüssel $K \in \mathbb{Z}_2^n$. Der Verschlüsselungsalgorithmus implementiert die Verschlüsselungsfunktion

$$\mathbf{Enc} : \mathbb{Z}_2^{2n} \to \mathbb{Z}_2^n, \quad (K, P) \mapsto P \oplus K.$$

Der Entschlüsselungsalgorithmus implementiert dieselbe Funktion

$$\mathbf{Dec} : \mathbb{Z}_2^{2n} \to \mathbb{Z}_2^n, \quad (K, C) \mapsto C \oplus K.$$

Theorem 4.3 *Das Vernam-One-Time-Pad ist perfekt sicher.*

Beweis Die Behauptung folgt aus Theorem 4.2. Auf dem Schlüsselraum wird die Gleichverteilung gewählt und für jeden Klartext P und jeden Schlüsseltext C gibt es genau einen Schlüssel K mit $C = P \oplus K$, nämlich $K = P \oplus C$. □

Das OTP wurde 1917 von Gilbert Vernam erfunden und patentiert. Aber erst 1949 formulierte Claude Shannon das Modell der perfekten Geheimhaltung und bewies, dass das OTP perfekt geheim ist.

Welche Bedeutung hat das OTP in der Praxis? Der Umstand, dass die Schlüssel dieselbe Länge haben müssen wie die Klartexte und nur einmal verwendet werden dürfen, schränkt die Praktikabilität des OTP ein. Dagegen ist der Verschlüsselungsalgorithmus sehr effizient. Er führt nur Additionen modulo 2 durch. Der große Vorteil des OTP ist seine perfekte Geheimhaltung. Angreifer können aus Chiffretexten niemals etwas über die verschlüsselten Klartexte lernen. Alle anderen Verschlüsselungsverfahren bieten nur einen zeitlich begrenzten Schutz. Wenn ein Angreifer also Schlüsseltexte abfängt und hinreichend lange aufbewahrt, kann er sie schließlich entschlüsseln. Wenn aber Daten elektronisch kommuniziert werden, die langfristig vertraulich bleiben müssen, wie zum Beispiel Gesundheitsdaten, ist möglicherweise langfristiger Schutz nötig. Das nächste Beispiel zeigt, wie das OTP eingesetzt werden kann.

Beispiel 4.4 Das Außenministerium eines Landes möchte mit einer Botschaft vertraulich kommunizieren. Weil die Nachrichten langfristig vertraulich bleiben sollen, bringt ein Kurier einmal im Jahr ein Speichermedium mit Zufallsbits in die Botschaft. Eine Kopie des Speichermediums bleibt im Außenministerium. Die Zufallsbits werden nach und nach eingesetzt, um mittels OTP die Vertraulichkeit der Kommunikation zwischen Botschaft und Außenministerium zu schützen. Bei jeder neuen Kommunikation werden neue Zufallsbits verwendet. Weil moderne Speichermedien eine sehr hohe Kapazität haben, kann auf diese Weise die perfekte Vertraulichkeit der gesamten Kommunikation innerhalb eines Jahres sichergestellt werden.

Im Beispiel 4.4 genügt der Schutz der Vertraulichkeit nicht. Die kommunizierten Daten müssen zum Beispiel auch vor Veränderung geschützt werden. Techniken dafür werden später beschrieben.

Das OTP erlaubt den langfristigen Schutz der Vertraulichkeit bei elektronischer Kommunikation. In vielen Fällen ist es aber auch nötig, die Vertraulichkeit gespeicherter Daten langfristig zu schützen, zum Beispiel in elektronischen Archiven. Dafür ist das OTP leider ungeeignet. Wird ein Archiv nämlich mit dem OTP geschützt, muss dazu ein Schlüssel verwendet werden, der genauso groß ist wie das Archiv. Die Vertraulichkeit dieses Schlüssels muss dann sichergestellt werden. Dies ist aber genauso aufwändig, wie das Archiv selbst zu schützen und nichts ist gewonnen. Die Vertraulichkeit gespeicherter Daten kann mit Hilfe von Secret-Sharing-Techniken erreicht werden. Dies wird in Kap. 15 besprochen.

4.3 Semantische Sicherheit

Abschn. 4.1 stellt ein mathematisches Modell für perfekte Geheimhaltung vor. In diesem Modell wird für jede Verschlüsselung ein neuer Schlüssel gewählt. Wird ein Schlüssel mehrfach verwendet, sind perfekt vertrauliche Verschlüsselungsverfahren dagegen unsicher. Dies zeigt das nächste Beispiel.

Beispiel 4.5 Angenommen, im OTP wird ein Schlüssel K mehrfach verwendet und dem Angreifer ist das Klartext-Schlüsseltext-Paar (P, C) bekannt, in dem C durch Verschlüsselung mit K aus P entsteht. Dann kann der Angreifer den Schlüssel K ermitteln. Er muss nur $P \oplus C = P \oplus P \oplus K = K$ berechnen.

In der Praxis werden aber fast immer Verschlüsselungsverfahren verwendet, die Schlüssel mehrfach benutzen. Für solche Verschlüsselungsverfahren ist das Modell der perfekten Vertraulichkeit ungeeignet. In den 1980er Jahren entwickelten Shafi Goldwasser und Silvio Micali das realistischere Modell der *semantischen Sicherheit* (siehe [32]). Es schwächt das Modell der perfekten Geheimhaltung ab. Semantische Sicherheit verlangt nicht mehr, dass die Verteilung auf den Chiffretexten unabhängig vom verschlüsselten Klartext sein muss, wie das bei perfekter Geheimhaltung gemäß Theorem 4.1 der Fall sein muss. Statt dessen genügt es für semantische Sicherheit, dass Angreifer die Verteilungen auf den Chiffretexten für verschiedene gleich lange Klartexte *mit ihren Möglichkeiten* nicht unterscheiden können. Das nächste Beispiel zeigt, das eine Anpassung der Sicherheitsanforderungen an die Möglichkeiten von Angreifern tatsächlich zu praktikablen Sicherheitsverfahren führt.

Beispiel 4.6 Ein Dieb stiehlt eine Kreditkarte. Er versucht, mit dieser Karte Geld abzuheben. Die Karte ist mit einer vierstelligen PIN geschützt. Es gibt 10000 solche PINs. Die kann der Dieb durchprobieren. Das ginge sehr schnell, wenn dem Durchprobieren nicht Grenzen gesetzt wären. Ein Fehlbedienungszähler sorgt dafür, dass die Karte nach drei Fehlversuchen unbrauchbar wird. Der Dieb hat also nur die Möglichkeit, dreimal zu raten. Die Chance, des Angreifers, bei dreimaligem Raten erfolgreich zu sein, ist $3/10000$. Nach gängiger Auffassung ist diese Wahrscheinlichkeit klein genug um die Kreditkarte zu schützen. Gleichzeitig sind vierstellige PINs praktikabel, weil Benutzer sich solche PINs merken können. Bei längeren PINs wird das schon schwieriger.

Wir stellen jetzt das mathematische Modell der semantischen Sicherheit eines symmetrischen Kryptosystems

$$\mathbf{E} = (\mathbf{K}, \mathbf{P}, \mathbf{C}, \mathbf{KeyGen}, \mathbf{Enc}, \mathbf{Dec})$$

vor. In diesem Modell ist es das Ziel des Angreifers, die Wahrscheinlichkeitsverteilung auf Chiffretexten zu zwei verschiedenen aber gleich langen Klartexten zu unterscheiden. Die Bedingung, dass die Klartexte gleich lang sein müssen, kommt daher, dass die Länge der Klartexte die Länge der Chiffretexte bestimmt. Es ist also im Allgemeinen nicht zu erreichen, dass Chiffretextverteilungen zu Klartexten unterschiedlicher Länge ununterscheidbar sind.

Wir beschreiben zunächst Angreifer auf die semantische Sicherheit von $\mathbf{E}$. Ein solcher Angreifer ist ein probabilistischer Algorithmus. Er bekommt einen Schlüsseltext C, der entweder die Verschlüsselung eines Klartextes P_0 und oder eines anderen Klartextes

P_1 ist. A entscheidet, welcher der Klartext verschlüsselt wurde, indem der Algorithmus entweder 0 oder 1 ausgibt. A ist erfolgreich, wenn die Antwort stimmt.

Wir beschreiben A genauer. Die beiden Klartexte P_0 und P_1 haben gleiche Länge. Ohne diese Forderung gäbe es keine semantisch sicheren Verschlüsselungsverfahren. Solche Chiffren sollen ja keine Unterscheidung zwischen den Verschlüsselungen zweier verschiedener Klartexte zulassen. Aber diese Unterscheidungsaufgabe ist zu leicht, wenn die Klartexte unterschiedliche Länge haben. Als nächstes wird festgelegt, dass A die Klartexte selbst wählt. Warum? Wenn ein Angreifer nicht zwischen Verschlüsselungen selbst gewählter Klartexte unterscheiden kann, kann er erst recht nicht zwischen Verschlüsselungen von Klartexten aus anderer Quelle unterscheiden. Er kann ja die Klartexte so wählen, wie es alle möglichen anderen Quellen tun. Jetzt müssen wir noch erklären, wie A die Verschlüsselung von P_0 oder P_1 erhält. Dazu hat A Zugriff auf ein *Orakel* $\mathbf{Enc}_{b,K}$. Dabei ist b ein Bit und K ein Schlüssel. Das Orakel verhält sich wie ein Unterprogramm, hat also eine Eingabe, nämlich (P_0, P_1), und eine Ausgabe, nämlich $C = \mathbf{Enc}(K, P_b)$. A sieht aber nicht, was das Orakel macht. Um zu zeigen, dass A Zugriff auf das Orakel $\mathbf{Enc}_{b,K}$ hat, schreiben wir $A^{\mathbf{Enc}_{b,K}}$.

Das nächste Beispiel zeigt einen Angreifer auf die semantische Sicherheit von Verschlüsselung im ECB-Mode, der wie oben beschrieben arbeitet. Die Idee dazu findet sich bereits in Beispiel 4.2.

Beispiel 4.7 Wie in Beispiel 4.2 verwenden wir eine Blockchiffre über dem Alphabet $\mathbb{Z}_2$, nennen n seine Blocklänge und betrachten das Verschlüsselungsverfahren, das Klartexte der Länge $2n$ im ECB-Mode verschlüsselt. Der Angreifer 4.1 hat Zugriff auf das Orakel $\mathbf{Enc}_{b,K}$ für einen Schlüssel K und ein Bit $b \in \mathbb{Z}_2$.

Angreifer 4.1 ($A^{\mathbf{Enc}_{b,K}}$)

(Angreifer auf die semantische Sicherheit von ECB)

Der Angreifer kennt die Blocklänge n der verwendeten Blockchiffre
$P_0 \leftarrow 1^{2n}$
$P_1 \leftarrow 0^n 1^n$
$C \leftarrow \mathbf{Enc}_{b,K}(P_0, P_1)$
if $C = XX$ mit $X \in \mathbb{Z}_2^n$ **then**
  return 0
else
  return 1
end if

Dieser Angreifer ist sehr effzient und gibt immer das richtige Bit b zurück. Er zeigt, dass ECB-Verschlüsselung nicht semantisch sicher ist.

Tab. 4.1 Sicherheitslevel

Schutz bis	Sicherheitslevel
2020	2^{96}
2030	2^{112}
2040	2^{128}
für absehbare Zukunft	2^{256}

Beispiel 4.7 beschreibt einen sehr einfachen Angreifer, der mit Wahrscheinlichkeit 1 entscheiden kann, welcher Klartext verschlüsselt wurde. Im allgemeinen haben es Angreifer aber nicht so leicht. Wir werden daher die Definition der semantischen Sicherheit noch etwas weiter entwickeln, um aussagen zu können, was Angreifer mit akzeptabler Laufzeit und Erfolgswahrscheinlichkeit sind.

Wir erläutern zuerst, was unter der Laufzeit eines Angreifers zu verstehen ist. Sie ist die Anzahl der Bit-Operationen, die der entsprechende probalistische Algorithmus im schlechtesten Fall benötigt. Dabei werden die Orakel-Aufrufe folgendermaßen berücksichtigt: Orakel verwenden eine Bit-Operation, um ihren Rückgabewert zu liefern. Der Algorithmus muss zusätzlich eine Bit-Operation pro Bit aufwenden, das er vom Rückgabewert liest. Um den schlechtesten Fall zu bestimmen, werden alle möglichen Münzwürfe im Angreiferalgorithmus und Rückgabewerte des Verschlüsselungsorakels berücksichtigt. Das nächste Beispiel bestimmt die Laufzeit des Angreifers aus Beispiel 4.7.

Beispiel 4.8 Wir bestimmen die Laufzeit, die der Angreifer aus Beispiel 4.7 verbraucht. Die Bestimmung der beiden Klartexte erfordert $4n$ Bit-Operationen. Der Angreifer ruft das Orakel einmal auf. Das Orakel gibt einen Schlüsseltext der Länge $2n$ zurück. Der Angreiferalgorithmus muss den gesamten Schlüsseltext lesen, um seine Struktur zu bestimmen. Dafür verwendet er nach obiger Konvention $2n$ Bit-Operationen. Die Analyse der Schlüsseltextes verwendet $O(n)$ Bit-Operationen. Damit ist die Laufzeit des Angreiferalgorithmus $O(n)$ und zwar für alle möglichen Münzwürfe und Rückgabewerte des Orakels.

Um aussagen zu können, dass es keine Angreifer mit akzeptabler Laufzeit gibt, wird häufig der Begriff des *Sicherheitslevels* verwendet. Dabei handelt es sich um eine natürliche Zahl l. Sicherheitslevel l bedeutet, dass alle Angriffe die nicht mehr als 2^l Operationen benötigen, erfolglos sind. Solange Angreifer also aufgrund von technologischen Beschränkungen höchstens 2^l Operationen ausführen können, kann es keine erfolgreichen Angriffe geben und das Verfahren ist sicher. Angreifer, die weniger als 2^l Operationen verwenden, werden also als realistisch eingeschätzt. Angreifer, die mehr Operationen verwenden, dagegen nicht. Welches Sicherheitslevel angemessen ist, hängt davon ab, wie lange die Sicherheit gewährleistet werden soll. Nach dem Mooreschen Gesetz verdoppelt sich nämlich die Geschwindigkeit von Computern alle 18 Monate. In Tab. 4.1 werden angemessene Sicherheitslevel angegeben. Die Werte wurden im Rahmen des EU-Projekts ECRYPT II ermittelt (siehe [73]).

Als nächstes erläutern wir, wie der Erfolg des Angreifers modelliert wird. Um diesen Erfolg zu bestimmen, verwenden wir ein *Experiment*. In einem solchen Experiment wählen wir die Eingaben für den Angreifer einschließlich des Orakels, auf das er Zugriff hat, zufällig. Im Experiment ist der Angreifer entweder erfolgreich oder nicht. Relevant für die Güte des Angriffs ist die Erfolgswahrscheinlichkeit. Für den Erfolg des Angreifers auf die semantische Sicherheit eines Verschlüsselungsverfahrens ist Experiment 4.1 relevant.

Experiment 4.1 ($\mathrm{Exp}_{\mathbf{E}}^{\mathrm{sem}}(A)$)

(Experiment, das über den Erfolg eines Angreifers auf die semantische Sicherheit des symmetrischen Kryptosystems **E** entscheidet)

$$
\begin{aligned}
&b \xleftarrow{\$} \mathbb{Z}_2\\
&K \leftarrow \mathbf{KeyGen}\\
&b' \leftarrow A^{\mathbf{Enc}_{b,K}}\\
&\textbf{if } b = b' \textbf{ then}\\
&\quad \textbf{return } 1\\
&\textbf{else}\\
&\quad \textbf{return } 0\\
&\textbf{end if}
\end{aligned}
$$

Um den Erfolg des Angreifers zu bestimmen, wird sein *Vorteil* (englisch: *advantage*) in Experiment 4.1 herangezogen, der folgendermaßen definiert ist:

$$\mathbf{Adv}_{\mathbf{E}}^{\mathrm{sem}}(A) = 2\Pr[\mathrm{Exp}_{\mathbf{E}}^{\mathrm{sem}}(A) = 1] - 1. \tag{4.7}$$

Diese Definition geht davon aus, dass ein Angreifer sein Rückgabe-Bit immer zufällig und gleichverteilt wählen kann. Dann ist seine Erfolgswahrscheinlichkeit $1/2$. Der Vorteil misst also die Wahrscheinlichkeit, die über $1/2$ hinausgeht. Wenn der Angreifer mit Wahrscheinlichkeit 1 den richtigen Wert zurückgibt, ist sein Vorteil 1. Rät der Angreifer nur, ist sein Vorteil 0. Offensichtlich hat der Angreifer aus Beispiel 4.7 den Vorteil 1.

Zum Schluss definieren wir semantische Sicherheit von symmetrischen Kryptosystemen.

Definition 4.2 Sei $T \in \mathbb{N}$ und $\varepsilon > 0$. Das Kryptosystem **E** bietet semantische (T, ε)-Sicherheit, wenn der Vorteil aller Angreifer gegen die semantische Sicherheit von **E**, die höchstens Laufzeit T haben, durch ε beschränkt ist.

Um diese Definition anwenden zu können, muss noch ein angemessenes Sicherheitslevel und der zulässige Vorteil festgelegt werden. Sicherheitslevel, die Schutz für bestimmte Zeitintervalle bieten, sind in Tab. 4.1 gezeigt. Eine mögliche Wahl für den maximal erlaub-

ten Vorteil ist $1/l$, wobei l das gewählte Sicherheitslevel ist. Soll ein Kryptosystem also bis zum Jahr 2020 sicher sein, darf der Vorteil von Angreifern, die höchstens die Laufzeit 2^{96} haben, nicht größer als $1/2^{96}$ sein.

4.4 Chosen-Plaintext-Sicherheit

Das Modell der semantischen Sicherheit berücksichtigt nur Ciphertext-Only-Angriffe. In Abschn. 3.4.2 wurden aber komplexere Angriffe beschrieben. In diesem Abschnitt beschreiben wir das Modell der *Chosen-Plaintext-Sicherheit (CPA-Sicherheit)*. In diesem Modell hat der Angreifer die Möglichkeit, sich Klartexte seiner Wahl verschlüsseln zu lassen. Genauer gesagt modellieren wir adaptive Chosen-Plaintext-Angriffe, in denen der Angreifer die zu verschlüsselnden Klartexte in Abhängigkeit von seinen vorherigen Berechnungen wählen darf. Das beschriebene Modell wird auch *Ununterscheidbarkeit bei Chosen-Plaintext-Angriffen* (Englisch: *Indistinguishability under Chosen-Plaintext-Attack (IND-CPA)* genannt. Das Ziel des Angreifers ist es nämlich, Wahrscheinlichkeitsverteilungen auf Klartexten zu unterschiedlichen Chiffretexten zu unterscheiden.

Wir gehen wieder davon aus, dass ein Verschlüsselungsverfahren

$$\mathbf{E} = (\mathbf{K}, \mathbf{P}, \mathbf{C}, \mathbf{KeyGen}, \mathbf{Enc}, \mathbf{Dec})$$

verwendet wird. Chosen-Plaintext-Angreifer auf $\mathbf{E}$ entsprechen Angreifern auf die semantische Sicherheit von $\mathbf{E}$. Sie haben aber zusätzlich zu dem Orakel $\mathbf{Enc}_{b,K}$, das einen von zwei Klartexten verschlüsselt, auch Zugriff auf ein Verschlüsselungsorakel $\mathbf{Enc}_K$, das eingegebene Klartexte verschlüsselt, ohne dabei den geheimen Schlüssel K preiszugeben. Chosen-Plaintext-Angreifer dürfen dieses Orakel so oft aufrufen, bis ihre erlaubte Laufzeit verbraucht ist. Die Definition des Erfolgs eines Chosen-Plaintext-Angreifers verwendet das Experiment 4.2. Der einzige Unterschied zum Experiment 4.1 besteht darin, dass der Angreifer zusätzlich Zugriff auf das Orakel $\mathbf{Enc}_K$ erhält, das er so oft aufrufen darf, wie es seine Zeitbeschränkung erlaubt.

Experiment 4.2 ($\mathrm{Exp}^{\mathrm{cpa}}_{\mathbf{E}}(A)$)

(Experiment, das über den Erfolg eines Chosen-Plaintext-Angreifers auf das Verschlüsselungsverfahren $\mathbf{E}$ entscheidet)

$$b \xleftarrow{\$} \mathbb{Z}_2$$
$$K \xleftarrow{\$} \mathbf{K}$$
$$b' \leftarrow A^{\mathbf{Enc}_{b,K}, \mathbf{Enc}_K}$$

if $b = b'$ **then**
  return 1
else
  return 0
end if

Der Vorteil des Angreifers ist definiert wie im Modell der semantischen Sicherheit:

$$\mathbf{Adv}^{\text{cpa}}_{\mathbf{E}}(A) = 2\Pr[\text{Exp}^{\text{cpa}}_{\mathbf{E}}(A) = 1] - 1. \tag{4.8}$$

Wir geben nun einige Beispiele für Verschlüsselungsverfahren, die gegen Chosen-Plaintext-Angriffe unsicher sind.

Beispiel 4.9 Wir zeigen, dass deterministische, zustandslose Verschlüsselungsverfahren nicht CPA-sicher sind. Angreifer 4.2 greift die CPA-Sicherheit des deterministischen zustandslosen Verschlüsselungsverfahrens **E** erfolgreich an.

Angreifer 4.2 ($A^{\mathbf{Enc}_{b,K},\mathbf{Enc}_K}$)

(Angreifer auf die CPA-Sicherheit von deterministischen, zustandslosen Verschlüsselungsverfahren)

Wähle zwei verschiedene, gleich lange Klartexte P_0 und
  P_1 aus **P**.
$C_0 \leftarrow \mathbf{Enc}_K(P_0)$
$C \leftarrow \mathbf{Enc}_{b,K}(P_0, P_1)$
if $C = C_0$ **then**
  return 0
else
  return 1
end if

Der Angreifer benötigt nur wenige Operationen: Die Auswahl zweier Klartexte, zwei Orakelaufrufe und ein Vergleich. Sein Vorteil ist 1, weil die Verschlüsselungsfunktionen $\mathbf{Enc}_K$, die zu einem deterministischen zustandslosen Verschlüsselungsverfahren gehören, injektiv sind.

Beispiel 4.10 Wir zeigen, dass Verschlüsselung im CBC-CTR-Mode nicht CPA-sicher ist. Verwendet wird eine Blockchiffre mit Blocklänge n und Verschlüsselungsfunktion E. Angreifer 4.3 greift die CPA-Sicherheit des CBC-CTR-Verschlüsselungsverfahrens **E** erfolgreich an.

Angreifer 4.3 ($A^{\mathbf{Enc}_{b,K},\mathbf{Enc}_K}$)

(Angreifer auf die CPA-Sicherheit von CBC-CTR-Verschlüsselungsverfahren)

Der Angreifer kennt die Blocklänge n der verwendeten Blockschiffre
$C_0 \leftarrow \mathbf{Enc}_K(0^n)$
$C \leftarrow \mathbf{Enc}_{b,K}(0^n, 0^{n-1}1)$
if $C = C_0$ **then**
  return 1
else
  return 0
end if

Auch dieser Angreifer benötigt nur wenige Operationen: zwei Orakelaufrufe und ein Vergleich. Außerdem ist sein Vorteil 1, weil bei jedem Aufruf der Verschlüsselungsfunktion der Zähler um eins erhöht wird. Das wird in Übung 4.4 gezeigt.

4.5 Chosen-Ciphertext-Sicherheit

In Abschn. 3.4.2 wurde ein weiterer Angriffstyp vorgestellt: der Chosen-Ciphertext-Angriff. In diesem Abschnitt wollen wir auch diesen Angriffstyp formal modellieren. Das entsprechende Modell heißt *Chosen-Ciphertext-Sicherheit*, *Indistinguishabilty under Chosen-Ciphertext-Attack (IND-CCA)* oder kurz *CCA-Sicherheit*. Sei $\mathbf{E} = (\mathbf{K}, \mathbf{P}, \mathbf{C}, \mathbf{KeyGen}, \mathbf{Enc}, \mathbf{Dec})$ ein Verschlüsselungsverfahren.

Wie im Chosen-Plaintext-Modell hat der Angreifer Zugriff auf die Orakel $\mathbf{Enc}_{b,K}$ und $\mathbf{Enc}_K$. Das erste Orakel darf er einmal auf ein Paar von Klartexten gleicher Länge anwenden; das zweite so oft es seine Laufzeitbeschränkung zulässt. Im Chosen-Ciphertext-Modell hat der Angreifer aber noch zusätzlich Zugriff auf das $\mathbf{Dec}_K$-Orakel. Es entschlüsselt gegebene Schlüsseltexte so oft es die Laufzeitbeschränkung des Angreifers zulässt. Könnte der Angreifer das Entschlüsselungsorakel auf alle Chiffretexte anwenden, hätte er leichtes Spiel. Er könnte das Verschlüsselungsorakel ein Paar von Klartexten verschlüsseln lassen und dann mit dem Entschlüsselungsorakel herausfinden, ob der linke oder der rechte Klartext von $\mathbf{Enc}_{b,K}$ verschlüsselt wurde. Ein Modell, das dies zulässt, ist nicht sinnvoll. In einem solchen Modell gibt es keine Sicherheit durch Verschlüsselung. Das Modell soll aber nur die Möglichkeit berücksichtigen, dass ein Angreifer vorübergehend Zugriff auf einen Entschlüsselungsmechanismus hat, aber diesen Mechanismus nicht für die Klartexte nutzen kann, die ihn primär interessieren. Daher gilt im Chosen-Ciphertext-Modell die Regel, dass das Entschlüsselungsorakel nur Chiffretexte entschlüsselt, die nicht zuvor durch eins der Orakel $\mathbf{Enc}_K$ oder $\mathbf{Enc}_{b,K}$ verschlüsselt wurden. Die weitere Formalisierung in Analogie zum Chosen-Plaintext-Modell überlassen wir dem Leser in Übung 4.8

Wir zeigen nun, dass im Chosen-Ciphertext-Modell CTR-Mode-Verschlüsselung unsicher ist.

Beispiel 4.11 Wir nehmen an, dass das Verschlüsselungsverfahren **E** entsteht, indem eine Blockschiffre der Blocklänge n im CTR-Mode angewendet wird. Angreifer 4.4 greift erfolgreich die Sicherheit des CRT-Mode an.

Angreifer 4.4 ($A^{\mathbf{Enc}_{b,K},\mathbf{Enc}_K,\mathbf{Dec}_K}$)

(Angreifer auf die CCA-Sicherheit von Verschlüsselung im CTR-Mode)

Der Angreifer kennt die Blocklänge n der
  verwendeten Blockchiffre
$C = \mathbf{Enc}_{b,K}(0^n, 1^n)$,
$P \leftarrow \mathbf{Dec}_K(C \oplus 1^n)$
if $P = 1^n$ **then**
  return 0
else
  return 1
end if

Der Angreifer hat lineare Laufzeit in der Blocklänge n, ist also sehr effizient. Wir zeigen, dass sein Vorteil 1 ist. Angenommen, $b = 0$. Dann ergibt die Verschlüsselung

$$C = 0^n \oplus E(K, \mathrm{IV} + 1), \tag{4.9}$$

wobei E die Verschlüsselungsfunktion der Blockchiffre bezeichnet. Die Anwendung des Entschlüsslungsorakels liefert

$$\begin{aligned} P &= C \oplus 1^n \oplus E(K, \mathrm{IV} + 1) \\ &= 0^n \oplus E(K, \mathrm{IV} + 1) \oplus 1^n \oplus E(K, \mathrm{IV} + 1) \\ &= 1^n. \end{aligned}$$

Nun sei $b = 1$. Dann gilt

$$C = 1^n \oplus E(K, \mathrm{IV} + 1) \tag{4.10}$$

und

$$\begin{aligned} P &= C \oplus 1^n \oplus \mathbf{Enc}(K, \mathrm{IV} + 1) \\ &= 1^n \oplus \mathbf{Enc}(K, \mathrm{IV} + 1) \oplus 1^n \oplus \mathbf{Enc}(K, \mathrm{IV} + 1) \\ &= 0^n. \end{aligned}$$

Der Angreifer gibt also immer den richtigen Wert von b aus.

4.6 Übungen

Übung 4.1 Zeigen Sie, dass die Verschiebungschiffre perfekt geheim ist, wenn die Klartexte einzelne Zeichen sind. Erklären Sie, warum, dies sinnvoll ist. Was passiert, wenn längere Texte verschlüsselt werden?

Übung 4.2 Betrachten Sie die lineare Blockchiffre mit Blocklänge n und Alphabet $\mathbb{Z}_2^n$. Wählen Sie auf dem Schlüsselraum aller Matrizen $A \in \mathbb{Z}_2^{(n,n)}$ mit $\det(A) \equiv 1 \bmod 2$ die Gleichverteilung. Ist dieses Kryptosystem perfekt geheim, wenn jeder neue Klartext der Länge n mit einem neuen Schlüssel verschlüsselt wird?

Übung 4.3 Beweisen Sie Theorem 4.1.

Übung 4.4 Zeigen Sie, dass der Angreifer 4.3 den Vorteil 1 hat.

Übung 4.5 Zeigen Sie, dass affin lineare Blockchiffren im IND-CPA-Modell unsicher sind. Geben Sie dazu einen IND-CPA-Angreifer für solche Blockchiffren an und bestimmen sie seine Laufzeit und seinen Vorteil.

Übung 4.6 Formalisieren Sie das IND-CCA-Modell. Definieren Sie dazu Angreifer im IND-CCA-Modell und seine Laufzeit. Definieren Sie auch den Vorteil eines Angreifers im IND-CCA-Modell.

Übung 4.7 Zeigen Sie, dass das Verschlüsselungsverfahren, das bei Verwendung von Blockchiffren im CBC-Mode mit zufälligem Initialisierungsvektor im IND-CCA-Modell unsicher ist. Geben Sie dazu einen IND-CCA-Angreifer für solche Blockchiffren an und bestimmen sie seine Laufzeit und seinen Vorteil.

Übung 4.8 Formalisieren Sie das INDCCA-Modell.