Aufgaben

4.1. AES ist im Jahr 2002 von der US-Standardisierungsbehörde NIST standardisiert worden.

1. Die Entstehungsgeschichte des AES unterscheidet sich stark von der des DES. Beschreiben Sie die Unterschiede bei der Entstehung der beiden Blockchiffren.
2. Was waren die wichtigsten Meilensteine des AES-Entstehungsprozesses?
3. Wie hieß der heutige AES-Algorithmus ursprünglich?
4. Wie heißen die beiden Erfinder des AES?
5. Welche Blöckgrößen und Schlüssellängen sind bei dem ursprünglich eingereichten Algorithmus möglich?

4.2. Innerhalb der AES-Rundenfunktion finden viele Berechnungen in endlichen Körpern statt. Die nachfolgenden Aufgaben beschäftigen sich mit Arithmetik in endlichen Körpern.

Erstellen Sie die Multiplikations- und Additionstafeln für den Primkörper $GF(7)$[1]. Die Multiplikationstafel ist eine quadratische (hier: 7×7) Tabelle, deren Zeilen und erste Spalten jeweils einem Körperelemente entsprechen. Die 49 Einträge sind alle möglichen Produkte, die gebildet werden können. Man beachte, dass die Tafel spiegelsymmetrisch bezüglich der Hauptdiagonalen ist, da die Multiplikation kommutativ ist. Die Additionstafel ist analog hierzu aufgebaut, enthält aber alle möglichen Summen der Körperelemente.

4.3. Erstellen Sie die Multiplikationstafel für den Erweiterungskörper $GF(2^3)$ mit dem irreduziblen Polynom $P(x) = x^3 + x + 1$. Es handelt sich um eine Tafel mit 8×8 Einträgen. Die Tafel kann manuell erstellt werden oder mittels eines Programms.

4.4. Berechnen Sie $A(x) + B(x) \mod P(x)$ in dem Körper $GF(2^4)$ mit dem irreduziblen Polynom $P(x) = x^4 + x + 1$.

1. $A(x) = x^2 + 1, B(x) = x^3 + x^2 + 1$
2. $A(x) = x^2 + 1, B(x) = x + 1$

Ändern sich die Additionsergebnisse, wenn ein anderes irreduzibles Polynom gewählt wird?

4.5. Berechnen Sie $A(x) \cdot B(x) \mod P(x)$ in $GF(2^4)$ mit dem irreduziblen Polynom $P(x) = x^4 + x + 1$.

1. $A(x) = x^2 + 1, B(x) = x^3 + x^2 + 1$
2. $A(x) = x^2 + 1, B(x) = x + 1$

Ändern sich die Additionsergebnisse, wenn ein anderes irreduzibles Polynom gewählt wird, z. B. $P(x) = x^4 + x^3 + 1$?

[1] In der Mathematik werden diese Tafeln auch Verknüpfungstafeln genannt und in der Gruppentheorie Cayley-Tafeln.

4.6. Berechnen Sie in $GF(2^8)$:

$$(x^4+x+1)/(x^7+x^6+x^3+x^2),$$

wobei das irreduzible Polynom verwendet wird, das auch für AES zum Einsatz kommt: $P(x) = x^8+x^4+x^3+x+1$. Man beachte, dass Tabelle 4.2 die multiplikativen Inversen aller Körperelemente enthält.

4.7. Wir betrachten den endlichen Körper $GF(2^4)$ mit dem irreduziblen Polynom $P(x) = x^4+x+1$. Berechnen Sie die Inverse für die beiden Elemente $A(x) = x$ und $B(x) = x^2+x$. Man kann die Inversen entweder durch Ausprobieren aller möglicher Körperelemente bestimmen, d. h. man führt eine vollständige Suche durch, oder mittels des erweiterten euklidischen Algorithmus. (Der Algorithmus wurde in diesem Kapitel allerdings nur kurz erwähnt.) Verifizieren die Lösungen durch Multiplikation mit A bzw. mit B.

4.8. Bestimmen Sie alle irreduziblen Polynome

1. mit Grad 3 über $GF(2)$,
2. mit Grad 4 über $GF(2)$.

Der beste Weg hierfür ist, alle Polynome mit einem kleineren Grad aufzulisten und zu überprüfen, ob sie das Polynom vom Grad 3 (bzw. Grad 4), welches man gerade untersucht, teilt.

4.9. Wir betrachten AES mit 128-Bit-Schlüssel. Was ist der Zustand nach der ersten Runde, wenn der Eingang der ersten Runde (nach der Verknüpfung mit k_0) aus 128 Einsen besteht und der Unterschlüssel k_1 ebenfalls nur aus Einsen besteht? Es ist hilfreich, wenn die Zwischen- und das Endergebnis in Form einer 4×4-Zustandsmatrix aufgeschrieben werden.

4.10. Diese Aufgabe beschäftigt sich mit der *Diffusionseigenschaft* von AES. Wir betrachten hierfür eine einzige Runde. Die vier 32-Bit-Worte $X = (x_0,x_1,x_2,x_3) =$ (`0x01000000`, `0x00000000`, `0x00000000`, `0x00000000`) seien der Eingabewert für AES. Die beiden ersten Unterschlüssel werden durch die Worte $W_0,\ldots,W_7$ gebildet, die die folgenden Werte haben:

$$\begin{aligned}
W_0 &= (\texttt{0x2B7E1516}),\\
W_1 &= (\texttt{0x28AED2A6}),\\
W_2 &= (\texttt{0xABF71588}),\\
W_3 &= (\texttt{0x09CF4F3C}),\\
W_4 &= (\texttt{0xA0FAFE17}),\\
W_5 &= (\texttt{0x88542CB1}),\\
W_6 &= (\texttt{0x23A33939}),\\
W_7 &= (\texttt{0x2A6C7605}).
\end{aligned}$$

Schreiben Sie alle Zwischenergebnisse nach der *ShiftRows*-, *SubBytes*- und *MixColumns*-Schicht in Form einer quadratischen Zustandsmatrix bei der Beantwortung der nachfolgenden Fragen. Die Aufgabe kann händisch oder mittels eines Computerprogramms gelöst werden.

1. Berechnen Sie den Ausgangszustand nach der ersten Runde für den Eingangswert X und den Unterschlüssel $W_0, \ldots, W_7$.
2. Was ist der Ausgangszustand, wenn der Eingang nur aus Nullen besteht, d. h. ein Bit von X seinen Wert ändert?
3. Wie viele der Ausgangsbits haben sich verändert? (Man beachte, dass hier nur eine einzige Runde betrachtet wird. In den nachfolgenden Runden werden schnell alle Zustandsbits durch die Änderung des einen Eingangsbits beeinflusst. Man spricht hier vom *Avalanche-Effekt*.)

4.11. Die MixColumn-Transformation des AES führt eine Matrix-Vektor-Multiplikation in dem endlichen Körper $GF(2^8)$ mit dem irreduziblen Polynom $P(x) = x^8 + x^4 + x^3 + x + 1$ durch. Es sei $b = (b_7 x^7 + \ldots + b_0)$ eines der vier Eingangsbytes der Matrix-Vektor-Multiplikation. Jedes Eingangsbyte wird mit den drei Konstanten 01, 02 und 03 multipliziert. Ziel ist es, die exakten Gleichungen auf Bitebene für die Multiplikation mit den drei Konstanten aufzustellen. Das Ergebnis der jeweiligen Konstanten-Multiplikation bezeichnen wir mit $d = (d_7 x^7 + \ldots + d_0)$, d. h. jedes Bit von d soll in Abhängigkeit der Bits von b ausgedrückt werden.

1. Wie lauten die Ausdrücke für das Berechnen der acht Bits $d = 01 \cdot b$?
2. Wie lauten die Ausdrücke für das Berechnen der acht Bits $d = 02 \cdot b$?
3. Wie lauten die Ausdrücke für das Berechnen der acht Bits $d = 03 \cdot b$?

Man beachte die AES-Konvention, bei der "01" das Polynom 1 bezeichnet, "02" steht für das Polynom x und "03" repräsentiert $x + 1$.

4.12. Wir untersuchen nun, wie viele logische Gatter benötigt werden, um die MixColumn-Operation auszuführen. Wir verwenden hierfür die Ergebnis von Aufgabe 4.11. Man beachte, dass ein XOR-Gatter mit zwei Eingängen äquivalent zu einer Addition in $GF(2)$ ist.

1. Wie viele XOR-Gatter werden jeweils benötigt, um eine Multiplikation mit den Konstanten 01, 02 and 03 in $GF(2^8)$ durchzuführen?
2. Wie viele Gatter braucht man, um die vollständige Matrix–Vektor-Multiplikation auszuführen?
3. Wie viele Gatter werden für die Hardware-Realisierung der gesamten Diffusionsschicht benötigt? Wir nehmen an, dass Byte-Permutationen keine Gatter benötigen.

4.13. Wir betrachten den ersten Teil der ByteSub-Operation (d. h. der AES-S-Box), in der eine Inversion in dem endlichen Körper stattfindet.

1. Bestimmen Sie die Inversen der Eingangsbytes 29, F3 und 01 unter Benutzung von Tabelle 4.2. Die drei Byte-Werte sind in hexadezimaler Notation gegeben.

2. Verifizieren Sie die Antworten, indem Sie die Lösungen mit dem jeweiligen Eingangsbyte in $GF(2^8)$ multiplizieren. Man beachte, dass jedes Byte zunächst als $GF(2^8)$-Polynom dargestellt werden muss. Das höchstwertige Bit (MSB) eines jeden Bytes ist hierbei der Koeffizient für x^7.

4.14. Berechnen Sie die Ausgabe der AES-S-Box, d. h. die ByteSub-Operation, für die Eingangsbytes 29, F3 und 01. Alle Bytes sind in hexadezimaler Notation gegeben.

1. Bestimmen Sie zunächst die Inversen der Eingangsbytes unter Benutzung von Tabelle 4.2, um die Zwischenwerte B' zu berechnen. Berechnen Sie danach den zweiten Teil der S-Box-Operation, die affine Abbildung.
2. Verifizieren Sie die Antworten mittels der S-Box-Tabelle 4.3.
3. Wie lautet der Ausgabewert für $S(00)$?

4.15. In dieser Aufgabe soll die Bit-Darstellung der Rundenkonstanten innerhalb des Schlüsselfahrplans *hergeleitet* werden. Zeigen Sie die Berechnung der folgenden Rundenkonstanten:

- $RC[8]$
- $RC[9]$
- $RC[10]$

4.16. In dieser Aufgabe betrachten wir einen Brute-Force-Angriff auf AES mit einem 192-Bit-Schlüssel. Gegeben sei ein ASIC (d. h. ein spezielles IC), welches $3 \cdot 10^7$ Schlüssel pro Sekunde überprüfen kann.

1. Wie lange dauert eine vollständige Schlüsselsuche im Durchschnitt, wenn 100.000 solcher ICs parallel eingesetzt werden? Setzen Sie das Ergebnis in Relation zu dem Alter des Universums, das mit 10^{10} Jahren angenommen wird.
2. Wir nehmen nun an, dass das Mooresche Gesetz auch in Zukunft noch gelten wird. Wie viele Jahre muss man warten, bis ein Brute-Force-Angriff im Durchschnitt 24 Stunden benötigt? Wir nehmen wiederum an, dass 100.000 ICs zur Verfügung stehen. Gehen Sie bei der Aufgabe von einer Verdoppelung der Rechenleistung all 18 Monate aus (vgl. auch Abschnitt 1.3.2).