Institut für Theoretische Informatik
Prof. Dr. Dennis Hofheinz
Übungsleiter: Thomas Agrikola

Stammvorlesung Sicherheit im Sommersemester 2017

Übungsblatt 1

Aufgabe 1. Gegeben ist der folgende Chiffretext. Ermitteln Sie den dazugehörigen Klartext. (Hinweise: Als Verschlüsselungsmechanismus wurde das Vigenère-Verfahren benutzt. Den Chiffretext finden Sie auch als Textdatei zum Herunterladen auf der Webseite zur Vorlesung.)

```
MZMTELFMPLWUFEWXTZPCJDQPBVUKSIEEQDIMIIDRLQNTPWFBZNYNTTQLMAFAGQADDYXOPLIDIWYXFIN
ISZBSCZUOPLIDMNGTTMCCEDTTCVMBEYGWACCPUMOMRWVZUPQLRCSRBSCTXITLXQFEGSDCDCSRICCGAO
YGDMJWCAAZOYWMSPWOMATQOUAXCXTWOFEPVZQYOPOCTQVOCROQPQOMATQOUELQXTMQGVEBEMTGJWGWT
IYYGOWFLXANEFIMBEYGWJFRMFQDAPQICRLMBEFIDMHCVQWEFIDAHFSIMCCEIICCSRQEGRQQRFXQMYDM
RBJDSGZNFEDTPQFMJMYKQELQKAIOCHUVEMFDMLIMZOEFIHQRCRQZPAMBPPPATMYHSTVSYPXJCMGWBSU
EUBPQWGJXGXFMOYRQENGTTMCRSFPPHSGZYYPANEFIEWNGIFGZDXTMLPXEESCRNIMZESMDFSIMORLMBE
FAMQECWOQAFIDELQIEAPLXUIWJCVCDREZWEFIDZPAVQIEGSZWQRLQDTEIZMCCGUXSCVFPHYMFMDALMT
WCRSMOZENJLEIFWMPIMSSGWOQAFIDMYASPMORAUKPUMFPVCCEWQBMRNPPIZBWCRSBSZENJLEIECNAIQ
LPBMZLPAVKXEGRSIDYQBTPULUKSRYDVPBSGBEMFQBSCTAMXRLQDTQMAVZDWUVMWEXNCCHFMYLCEWYCR
OZJNXQLLAGAZOGRSBZRLQSPWAAZOCQUTJRLQNTPWFVLKIANECRZGDMREETDINIMZESMYCZQZPVTXITL
IPBSCQQBSMHTMFQIPAESHUMDMJNIMZESMDLSFMDPIHMLJXTIEFITIOSWQLEFIYMEFSPTLRIDXFZPUAS
CHNGVYWUAVGEZLDSKSMDRXTIEFITIOZIQVFQMZOEFIYMEFSPIDCEDTJYWQQRFXQMYDSGZEWWUF
```

Lösungsvorschlag zu Aufgabe 1. Wir interpretieren den Chiffretext C als Zahlenfolge $C = (C_i)_{i=1}^n \in \{0, \dots, 25\}^n$, wobei $(A \leftrightarrow 0, B \leftrightarrow 1, \dots, Z \leftrightarrow 25)$ gilt. Um aus Vigènere-Chiffraten den Klartext zu ermitteln, gehen wir wie folgt vor:

(a) Wir raten die Länge ℓ des Schlüssels $K = K_1 K_2 \dots K_\ell \in \{0, \dots, 25\}^\ell$ (oder wir nutzen die Kasiski-, Friedmann- oder die Autokorrelations-Methode, die einen Kandidaten für m ausgeben). In unserem Fall raten wir $\ell = 5$.

(b) Wir zerlegen das Chiffrat in ℓ Teilchiffrate $C^{(i)} := (C_i, C_{i+\ell}, C_{i+2\ell}, \dots)$ für $i \in \{1, \dots, \ell\}$. Jedes Teilchiffrat $C^{(i)}$ ist nun einfach eine Caesar-Verschlüsselung mit Schlüsselkomponente K_i. Unser Ziel ist es diese ℓ getrennten Caesar-Chiffren durch eine Frequenzanalyse zu brechen.

(c) Die Idee hinter der Frequenzanalyse der Teilchiffrate ist, dass ein Buchstabe im Klartext (beispielsweise `A`) stets auf den gleichen Buchstaben im Teilchiffrat abgebildet wird (beispielsweise `K`). Da Buchstaben in natürlicher Sprache nicht gleichverteilt auftreten, können wir so aus dem Chiffrat Rückschlüsse auf den Klartext und sogar den verwendeten Schlüssel ziehen. Im Englischen ist die erwartete Buchstabenverteilung beispielsweise wie folgt:

```
A: ********************************
B: ******
C: ***********
D: ******************
E: **************************************************
F: *********
G: ********
H: ************************
I: ****************************
J: *
K: ***
L: ****************
M: **********
N: ***************************
O: *******************************
P: ********
Q:
R: ************************
S: *************************
T: ************************************
U: ***********
V: ****
W: *********
X: *
Y: ********
Z:
Erwartete Buchstabenhäufigkeit im Englischen (Quelle: Wikipedia).
```

Im Deutschen sieht es etwas anders aus:

```
A: ***************************
B: ********
C: ************
D: ********************
E: ************************************************************************
F: *******
G: ************
H: ********************
I: *******************************
J: *
K: *****
L: **************
M: **********
N: ***************************************
O: **********
P: ***
Q:
R: ****************************
S: ******************************
T: *************************
U: ******************
V: ***
W: ********
X:
Y:
Z: *****
Erwartete Buchstabenhäufigkeit im Deutschen (Quelle: Wikipedia).
```

Die Frequenzanalyse des ersten Teilchiffrats $C^{(1)}$ sieht nun wie folgt aus:

```
A: ***********
B:
C: ******
D: **
E: ********************************
F: ******
G: ******************
H: ***********
I: ************************************************************
J: ****
K: ******
L: ***********************
M: *************************
N:
O: ********
P: *************
Q: ******
R: ****************************
S: ************************************
T: ***************
U:
V: ***************
W: ****************************
X: *********************************
Y: ********
Z: ****
```

Der häufigste Buchstabe ist das `I`. Wahrscheinlich wurde also beim Verschlüsseln das `E` auf `I` abgebildet. Dies würde bedeuten, dass für die erste Schlüsselkomponente $K_1 = \texttt{I} - \texttt{E} = \texttt{E}$ gilt. So gehen wir nun bei allen Teilchiffraten vor und erhalten als Kandidaten für den Schlüssel $K = \texttt{EMIAY}$. Versuchen wir das Chiffrat mit diesem Schlüssel zu entschlüsseln, erhalten wir einen Text, der fast wie natürliches Englisch aussieht. Allerdings scheint das Teilchiffrat für $K_4 = \texttt{A}$ noch nicht richtig entschlüsselt zu sein. Werfen wir einen Blick auf die entsprechende Buchstabenverteilung:

```
A: ****
B:
C: **********************
D: *********************************
E: ************************************************************
F: *************
G:
H: ******
I:
J: *********
K:
L: ***********************
M: *************
N: *************
O: ******************************
P: ************************************************
Q: ******
R: *********
S: ******************************
T: *************************
U:
V: ******
W: *************
X: ******
Y: **********************
Z: *************
```

Wir vermuten nun, dass das E auf das zweithäufigste Zeichen – also das P – abgebildet wurde. Damit ergäbe sich L als vierte Schlüsselkomponente und als Schlüssel insgesamt $K =$ EMILY. Tatsächlich liefert dieser Schlüssel (nach Hinzufügen von Groß-/Kleinschreibung, Leer- und Satzzeichen und Ersetzen von Zahlwörtern,) den Klartext:

In 1863 Friedrich Kasiski was the first to publish a successful general attack on the Vigenère cipher. Earlier attacks relied on knowledge of the plaintext, or use of a recognizable word as a key. Kasiski's method had no such dependencies. Kasiski was the first to publish an account of the attack, but it is clear that there were others who were aware of it. In 1854, Charles Babbage was goaded into breaking the Vigenère cipher when John Hall Brock Thwaites submitted a "new" cipher to the Journal of the Society of the Arts. When Babbage showed that Thwaites' cipher was essentially just another recreation of the Vigenère cipher, Thwaites challenged Babbage to break his cipher encoded twice, with keys of different length. Babbage succeeded in decrypting a sample, which turned out to be the poem "The Vision of Sin", by Alfred Tennyson, encrypted according to the keyword Ëmily", the first name of Tennyson's wife. Babbage never explained the method he used. Studies of Babbage's notes reveal that he had used the method later published by Kasiski, and suggest that he had been using the method as early as 1846. (Quelle: http://en.wikipedia.org/wiki/Vigenère_cipher)

Aufgabe 2. Wir wissen, dass die Blockchiffre $(\mathsf{E}, \mathsf{D})\colon \{0,1\}^8 \times \{0,1\}^4 \to \{0,1\}^4$ bei Eingabe eines festen Schlüssels K_0 eine Eingabe M wie folgt auf eine Ausgabe $C := \mathsf{E}(K_0, M)$ abbildet:

M	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
C	0000	0001	1001	1110	1111	1011	0111	0110	1101	0010	1100	0101	1010	0100	0011	1000

Verschlüsseln Sie die Klartexte $M_1 =$ 0100 0111 0100 0001 und $M_2 =$ 0100 1101 0100 0001 unter K_0 in den Betriebsmodi Electronic Code Book (ECB), Cipher Block Chaining (CBC), und Counter Mode (CTR). Wählen Sie, falls nötig, einen geeigneten Initialisierungsvektor. Worauf sollte bei der Wahl eines Initialisierungsvektors IV in den einzelnen Modi, geachtet werden? (Beispielsweise: Falls E ununterscheidbar von einer Zufallsfunktion ist, wie sollte IV im CBC-Modus oder CTR-Modus gewählt werden, um passive Sicherheit zu gewährleisten?)

Lösungsvorschlag zu Aufgabe 2. Eine Verschlüsselung mit Initialisierungsvektor $IV_1 :=$ 0000 (für Klartext 1) bzw. $IV_2 :=$ 0101 (für Klartext 2) im CBC- und CTR-Modus (und ohne Initialisierungsvektor im ECB-Modus) sieht wie folgt aus:

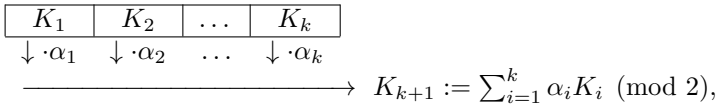
$M_{1,2}$	0100 0111 0100 0001	0100 1101 0100 0001
ECB	1111 0110 1111 0001	1111 0100 1111 0001
CBC	1111 1101 0010 1110	0001 1010 0011 1001
CTR	0101 1110 1010 1110	0011 1011 1001 0011

Eine notwendige Bedingung, um passive Sicherheit im CBC-Modus zu gewährleisten, ist die zufällige und gleichverteilte Wahl des Initialisierungsvektors für jeden Verschlüsselungsvorgang. (Um sich gegen aktive Angreifer abzusichern, müssen jedoch andere Maßnahmen ergriffen werden.)

Aufgabe 3. Aus der Vorlesung ist bekannt, dass One-Time-Pad-Chiffrate verformbar sind. (Ein Chiffrat $C := M \oplus K$, mit Klartext M und Schlüssel K, können wir verformen, indem wir $C' := C \oplus X$, für ein beliebiges X, berechnen. Bei der Entschlüsselung wird daraus $M' := D(K, C') = C' \oplus K = (C \oplus X) \oplus K = M \oplus X$.) Das im HEX-Format gegebene Chiffrat "DF C0 71 42 8A 90 17 0E 14 12" verschlüsselt das Wort "COMPLEXIFY". Verformen Sie dieses, sodass bei der Entschlüsselung der Klartext "DOKTORMETA" entsteht. (Hinweis: Die Buchstaben sind im ASCII-Format kodiert.)

Lösungsvorschlag zu Aufgabe 3. Nach einer Verformung $C' := C \oplus X$, für ein One-Time-Pad-Chiffrat C und beliebiges X, gilt $C' = (M \oplus X) \oplus K$, für einen Klartext $M = D(K, C)$. Um den Klartext $M :=$ `43 4F 4D 50 4C 45 58 49 46 59` (entspricht nach der ASCII-Tabelle und HEX-codiert dem Wort "COMPLEXIFY") in $M' =$ `44 4F 4B 54 4F 52 4D 45 54 41` (entspricht "DOKTORMETA") zu ändern, führen wir zuerst eine XOR-Verknüpfung $X := M \oplus M' =$ `07 00 06 04 03 17 15 0C 12 18` durch. Anschließend berechnen wir $C' := C \oplus X =$ `D8 C0 77 46 89 87 02 02 06 0A` und erhalten damit $M' = D(K, C') =$ `44 4F 4B 54 4F 52 4D 45 54 41`, was dem Wort "DOKTORMETA" enstpricht. (Hinweis: Auch der Schlüssel K ist bekannt; denn es gilt $K = C \oplus M =$ `9C 8F 3C 12 C6 D5 4F 47 52 4B`.)

Aufgabe 4. Aus der Vorlesung sind Linear-Feedback Shift Registers (LFSRs) als Beispiel für Pseudozufallsgeneratoren für Stromchiffren bekannt. Ihr Zustand besteht aus k Bits $K_1, \ldots, K_k$. Bei einem Zustandsupdate wird ein neues Bit K_{k+1} wie folgt berechnet:



wobei $\alpha_1, \ldots, \alpha_k \in \{0, 1\}$ geheim sind. Das erste Bit des jeweils aktuellen Zustands $c_1 := K_1$ wird ausgegeben, der aktualisierte Zustand ist $K_2, \ldots, K_{k+1}$.

Geben Sie einen Angriff auf dieses Verfahren an. **Genauer:** Geben Sie eine Möglichkeit an, gegeben eine Sequenz von $\mathbf{O}(k)$ Ausgabe-Bits des Zufallsgenerators, mindestens ein weiteres Ausgabe-Bit vorherzusagen. (Das sollte bei einem guten Zufallsgenerator nicht funktionieren.) Wie viele Ausgabe-Bits müssen wir betrachten, damit ein solcher Angriff immer gelingt?

Lösungsvorschlag zu Aufgabe 4.
Aufgabe: gegeben eine Sequenz $c_1, \ldots, c_{2k}$ soll mindestens ein weiteres Bit vorhergesagt werden. Ohne Einschränkung nennen wir die Zustandsbits, die der Zufallsgenerator unmittelbar vor Ausgabe des ersten Bits unserer gegebenen Sequenz hat, $K_1, \ldots, K_k$. Die ersten k Bits, die der Zufallsgenerator ausgibt, entsprechen seinem internen Zustand $c_1 := K_1, \ldots, c_k := K_k$.

Die Berechnung eines neuen Zustandsbits K_{k+1} bei einem Zustandsupdate kann als Produkt

$$K_{k+1} = (K_1, \ldots, K_k) \cdot \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix}$$

aufgefasst werden. Die Bits $(c_{k+1}, \ldots)$, die der Zufallsgenerator ab dem $k+1$-ten ausgegebenen Bit produziert, haben folgende Eigenschaft:

$$\underbrace{\begin{pmatrix} K_1 & \ldots & K_k \\ K_2 & \ldots & K_{k+1} \\ \vdots & & \vdots \end{pmatrix}}_{=:A} \cdot \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix} = \begin{pmatrix} c_{k+1} \\ \vdots \\ c_{2k} \end{pmatrix}.$$

Zudem gilt $c_{k+1} = K_{k+1}, \ldots, c_{2k} = K_{2k}$.

Offenbar ist es einfach die Ausgabebits des Zufallsgenerators vorherzusagen, wenn die Gewichte α_i bekannt sind (der aktuelle Zustand wird ohnehin Bit für Bit ausgegeben). Ist die entstehende Matrix A invertierbar in $\left(\mathbb{Z}/2\mathbb{Z}\right)^{k \times k}$, so können die Gewichte α_i durch $A^{-1} \cdot (c_{k+1}, \ldots, c_{2k})^T$ berechnet werden.

Ist A jedoch nicht invertierbar, so existiert ein (kleinstes) j, so dass die j-te Zeile Z_j von A in der linearen Hülle der ersten $j-1$ Zeilen $Z_1, \ldots, Z_{j-1}$ von A liegt. Also gilt $Z_j = \sum_{i=1}^{j-1} \gamma_i \cdot Z_i$ für Koeffizienten $\gamma_i \in \mathbb{Z}/2\mathbb{Z}$. Die Koeffizienten γ_i können z.B. durch Lösen eines linearen Gleichungssystems berechnet werden. Dann kann das Bit c_{k+j}, das der Zufallsgenerator als nächstes ausgeben wird, wie folgt vorhergesagt

werden:

$$c_{k+j} = (Z_j) \cdot \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix} = \left(\sum_{i=1}^{j-1} \gamma_i \cdot Z_i \right) \cdot \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix} = \sum_{i=1}^{j-1} \gamma_i \cdot \underbrace{Z_i \cdot \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix}}_{=c_{k+i}}.$$