Übungen - Angewandte Kryptologie

Alexander Weigl Martin Kock Timmie Yuen
Daniel Wahlscheid Thomas Schrader

July 24, 2011

Contents

1. Übung 1

1.1. Aufgabe 1

Chiffre	**Alphabet**	**Geheimtext**	**Schlüsselraum**	**Länge**
Caesar	$\{A, \cdots, Z\}$	$\{A, \cdots, Z\}$	$\{3\}, \{1, \cdots, 25\}$	4,64
OTP	$\{0,1\}$	$\{0,1\}$	$\{0,1\}^*$	∞
DES	$\{0,1\}$	$\{0,1\}$	$\{0,1\}^{56}$	56

1.2. Aufgabe 2

a. Beweis: Äquivalenzrelation: $\equiv_n$

Reflexivität $\forall x : x \equiv_n x$

$$x \mod n \equiv r = x \mod n \Rightarrow x \equiv_n x \tag{1}$$

#

Symmetrie $\forall x, y : x \equiv_n y \Rightarrow y \equiv_n x$

$$\begin{aligned} x \equiv_n y \mod n \Rightarrow x \mod n = r = y \mod n \\ \Rightarrow y \mod n = x \mod n \Rightarrow y \equiv_n x \end{aligned}$$

#

Transitiviät $\forall x, y, z : x \equiv_n y, y \equiv_n z => x \equiv_n z$

$$\begin{aligned} n.V.x \mod n = r_x, \\ y \mod n = r_y \\ z \mod n = r_z \wedge r_x = r_y, r_y = r_z \Rightarrow \\ r_x = r_z \Rightarrow x \equiv_n z \mod n \end{aligned}$$

#

b.

z. Z. $[i]_n + [j]_n = [i + j]_n$

$$\text{Sei } a, b \in \mathbb{Z} \Rightarrow a = q_a n + r_a, b = q_b n + r_b$$

$$\begin{aligned}
&\Rightarrow a \in [r_a]_n, b \in [r_b]_n \\
&\Rightarrow [r_a]_n + [r_b]_n = \{\forall i : in(r_a + r_b)\} \\
&\Rightarrow a + b = q_a n + r_a + q_b n + r_b \\
&\equiv_n n(q_a + q_b) + r_a + r_b \\
&\equiv_n r_a + r_b \Rightarrow [r_a + r_b]_n
\end{aligned}$$

z. Z. $[i]_n \cdot [j]_n = [i \cdot j]_n$

$$\text{Sei } a, b \in \mathbb{Z} \Rightarrow a = q_a n + r_a, b = q_b n + r_b$$

$$\begin{aligned}
&\Rightarrow a \in [r_a]_n, b \in [r_b]_n \\
&\Rightarrow [r_a]_n * [r_b]_n = \{\forall i : in(r_a * r_b)\} \\
&\Rightarrow a * b \equiv_n (q_a n + r_a) * (q_b n + r_b) \\
&\equiv_n q_a q_b n + q_a n r_b + q_b n r_a + r_a * r_b \\
&\equiv_n n(q_a q_b + q_a r_b + q_b r_a) + r_a * r_b \\
&\equiv_n r_a * r_b = [r_a + r_b]_n
\end{aligned}$$

1.3.

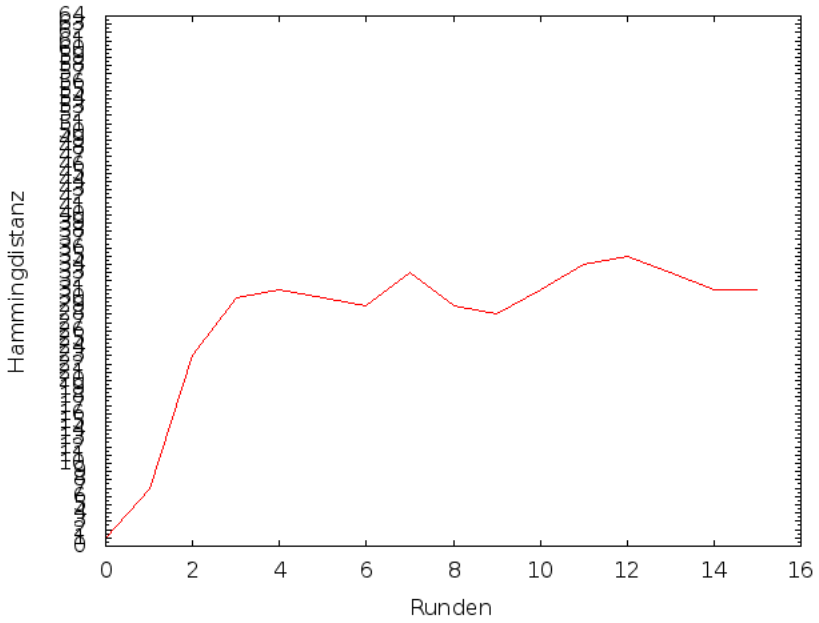
a.

$$\begin{aligned}
(23.145 \cdot = 12.479 + 14.543) \cdot \quad \mod 9 &\equiv_9 \\
(6 \cdot 5 + 8) \cdot 5 &\equiv_9 3 \\
8 \cdot 5 &\equiv_9 \\
2 \cdot 5 \equiv_9 10 = 1 \quad \mod 9
\end{aligned}$$

b.

$$\begin{aligned}
123 \cdot (123.983 \cdot 789.345 + 676.345) mod 11 &\equiv_{11} \\
2 \cdot (2 \cdot 7 + 10) &\equiv_{11} \\
2 \cdot (14 + 10) \equiv_{11} 4 \quad \mod 11
\end{aligned}$$

1.4.



Round k	$HD(M_k, M_k)$	$HD(M_{1_k}, M_{1_k})$	$HD(M_{1_k}, M_{1_k})$
0	1	39	35
1	7	35	31
2	23	32	33
3	30	36	35
4	31	37	34
5	30	31	34
6	29	26	30
7	33	26	26
8	29	29	26
9	28	29	34
10	31	25	36
11	34	28	35
12	35	30	38
13	33	29	37
14	31	32	38
15	31		

2. Übung 2

2.1. Aufgabe 1

a. Verschiebechiffren

$$E_1 : z \mapsto (z + k_1) \mod n \tag{2}$$
$$E_2 : z \mapsto (z + k_2) \mod n \tag{3}$$

Dann wäre die Verkettung $E_2 \circ E_1$:

$$E_2 \circ E_1 = E_2(E_1(z)) = (((z + k_1) \mod n) + k_2) \mod n \tag{4}$$
$$= z + \underbrace{k_1 + k_2}_{k_3} \mod n \tag{5}$$
$$= z + k_3 \mod n = E_3(z) \tag{6}$$

Wir folgern daraus, dass eine Verkettung von zwei Verschiebechiffren keine zusätzlichen Gewinn bringt.

b. Multiplikative Chiffren

$$E_1 : z \mapsto (z \cdot t_1) \mod n \tag{7}$$
$$E_2 : z \mapsto (z \cdot t_2) \mod n \tag{8}$$

Dann wäre die Verkettung $E_2 \circ E_1$:

$$E_2 \circ E_1 = E_2(E_1(z)) = (((z \cdot t_1) \mod n) \cdot t_2) \mod n \tag{9}$$
$$= z \cdot \underbrace{t_1 \cdot t_2}_{t_3} \mod n \tag{10}$$
$$= z \cdot t_3 \mod n = E_3(z) \tag{11}$$

Wir folgern daraus, dass eine Verkettung von zwei Multiplikativen Chiffren keine zusätzlichen Gewinn bringt.

c. Tauschchiffren

$$E_1 : z \mapsto (z \cdot t_1 + k_1) \mod n \tag{12}$$
$$E_2 : z \mapsto (z \cdot t_2 + k_2) \mod n \tag{13}$$

Dann wäre die Verkettung $E_2 \circ E_1$:

$$E_2 \circ E_1 = E_2(E_1(z)) = (((z \cdot t_1 + k_1) \mod n) \cdot t_2 + k_2) \mod n \tag{14}$$
$$= z \cdot \underbrace{t_1 \cdot t_2}_{t_3} + \underbrace{k_1 \cdot t_2 + k_2}_{k_3} \mod n \tag{15}$$
$$= z \cdot t_3 + k_3 \mod n = E_3(z) \tag{16}$$

Wir folgern daraus, dass eine Verkettung von zwei Tauschchiffren keine zusätzlichen Gewinn bringt.

2.2. Aufgabe 2

Berechnen Sie die multiplikativen Inverse zu $3,\ 5$ **und** 22 **in** Z_23**.**

$$\begin{aligned} 23 &= 7 \cdot 3 + 2 \\ 3 &= 1 \cdot 2 + 1 \\ 2 &= 2 \cdot 1 + 0 \end{aligned}$$

$$\begin{aligned} 1 &= 3 - 2 \\ 1 &= 3 - (23 - 7 \cdot 3) = \\ 1 &= \underline{8} \cdot -1 \cdot 23 \end{aligned}$$

$$\begin{aligned} 23 &= 1 \cdot 15 + 8 \\ 15 &= 1 \cdot 8 + 7 \\ 8 &= 1 \cdot 7 + 1 \\ 7 &= 7 \cdot 1 + 0 \end{aligned}$$

$$\begin{aligned} 1 &= 8 - 7 \\ 1 &= (23 - 15) - (15 - 8) \\ 1 &= (23 - 15) - (15 - (23 - 15)) \\ 1 &= 23 - 15 - 15 + 23 - 15 \\ 1 &= \underbrace{-3}_{20} \cdot 15 + 2 \cdot 23 \end{aligned}$$

$$\begin{aligned} 23 &= 1 \cdot 22 + 1 \\ 22 &= 22 \cdot 1 + 0 \end{aligned}$$

$$1 = 1 \cdot 23 \underbrace{-1}_{22} \cdot 22$$

Berechnen Sie die multiplikativen Inversen zu $3,\ 15$ **und** 22 **in** Z_{24}.

$$\begin{aligned} 24 &= 3 \cdot 8 + 0 \\ &\Rightarrow \neg\exists \text{ multiplikatives Inverses} \end{aligned}$$

$$\begin{aligned} 24 &= 1 \cdot 15 + 9 \\ 15 &= 1 \cdot 9 + 6 \\ 9 &= 1 \cdot 6 + 3 \\ 6 &= 2 \cdot 3 + 0 \\ &\Rightarrow \neg\exists \text{ multiplikatives Inverses} \end{aligned}$$

$$\begin{aligned} 24 &= 1 \cdot 22 + 2 \\ 22 &= 11 \cdot 2 + 0 \\ &\Rightarrow \neg\exists \text{ multiplikatives Inverses} \end{aligned}$$

Zeigen Sie, dass $(n-1)$ **in** Z_n **bzgl. der Multiplikation zu sich selbst invers ist.**

$$\begin{aligned} (n-1) \cdot (n-1) &\equiv_n n^2 - 2n + 1 && (17) \\ &\equiv_n 1 \mod n && (18) \end{aligned}$$

2.3. Aufgabe 3

Alphabet: `ABCDEFGHIJKLMNOPQRSTUVWXYZ` Alphabet: `UEBRDNWOLKMSIFHTACGJPQVXYZ`

Wind Nord-Ost, Startbahn null-drei,
Bis hier hör ich die Motoren.
Wie ein Pfeil zieht sie vorbei,
Und es dröhnt in meinen Ohren.
Und der nasse Asphalt bebt,
Wie ein Schleier staubt der Regen,
Bis sie abhebt und sie schwebt
Der Sonne entgegen.
Über den Wolken

Muß die Freiheit wohl grenzenlos sein.
Alle Ängste, alle Sorgen, sagt man,
Blieben darunter verborgen und dann
Würde, was hier gross und wichtig erscheint,
Plötzlich nichtig und klein.

2.4. Aufgabe 4

a.

Die Playfair-Verschlüsselung stellt eine Substitution für Buchstaben-Paare dar. Es handelt sich um eine bigraphische monoalphabetische Methode. Ähnlich wie bei der einfachen (monographischen) Buchstabensubstitution, beruhen Methoden zur Entzifferung von Playfair im Wesentlichen auf einer Analyse der Häufigkeitsverteilung hier der Buchstabenpaare (Bigramme). In der deutschen Sprache beispielsweise sind die Bigramme "er", "en" und "ch" sehr häufig. Im Beispieltext fallen die "Doppler" (also Bigramm-Wiederholungen) ME...ME, IK...IK, QC...QC und TE...TE sowie die "Reversen" (Wiederholung eines umgedrehten Bigramms) CQ...QC auf, die sich in gleicher Weise im englischen Klartext wiederfinden. Da kein Buchstabe mit sich selbst gepaart wird, gibt es nur 600 (25×24) mögliche Buchstabenkombinationen, die substituiert werden. Überdies gibt es eine Reihe von Symmetrien, die teilweise schon am obigen Beispieltext erkannt werden können. So hilft der erwähnte Klartext-Geheimtext-Zusammenhang EL $\leftrightarrow$ CQ und LE $\leftrightarrow$ QC beim Bruch des Textes. Ist nämlich ein Bigramm geknackt, dann ist auch sofort das reverse (umgedrehte) Bigramm bekannt. In den Fällen des Überkreuz-Schrittes gibt es darüber hinaus noch weitere Beziehungen zwischen den vier auftretenden Buchstaben in der Art (vgl. beispielsweise obere linke Ecke des Quadrats) DC $\leftrightarrow$ EB, CD $\leftrightarrow$ BE, EB $\leftrightarrow$ DC sowie BE $\leftrightarrow$ CD, die der Angreifer zur Entzifferung ausnutzen kann. Ferner hat auch die geschilderte Methode zur Erzeugung des Playfair-Quadrats Schwächen, denn es endet häufig – wie auch im Beispiel – auf "XYZ". Die Playfair-Verschlüsselung ist somit weit entfernt von einer allgemeinen bigraphischen Methode mit völlig willkürlicher Zuordnung der Buchstabenpaare und stellt in der heutigen Zeit kein sicheres Verschlüsselungsverfahren mehr dar. So lassen sich mit modernen Mitteln auch relativ kurze Playfair-Texte in sehr kurzer Zeit brechen.

2.5. Aufgabe 5

Herleitung des Gleichungssystems:

$$Hx_1 + Ix_2 = \ddot{A} \qquad Lx_1 + Lx_2 = U \tag{19}$$

$$Hx_3 + Ix_4 = U \qquad Lx_3 + Lx_4 = K \tag{20}$$

$$\tag{21}$$

$$\begin{pmatrix} 7 & 8 & 0 & 0 \\ 0 & 0 & 7 & 8 \\ 11 & 11 & 0 & 0 \\ 0 & 0 & 11 & 11 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 26 \\ 20 \\ 20 \\ 10 \end{pmatrix}$$

$$\begin{pmatrix} 7 & 8 & 0 & 0 & 26 \\ 0 & 0 & 7 & 8 & 20 \\ 11 & 11 & 0 & 0 & 20 \\ 0 & 0 & 11 & 11 & 10 \end{pmatrix} \leftarrow \begin{pmatrix} 7 & 8 & 0 & 0 & 26 \\ 11 & 11 & 0 & 0 & 20 \\ 0 & 0 & 7 & 8 & 20 \\ 0 & 0 & 11 & 11 & 10 \end{pmatrix} \begin{matrix} \\ \\ \\ \end{matrix} \begin{matrix} \\ \cdot -1 \\ \\ \cdot -1 \end{matrix} \begin{matrix} \\ + \\ \\ + \end{matrix} \begin{matrix} \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \end{matrix} \begin{matrix} \\ \mid \cdot 7^{-1} = 25 \\ \mid \cdot 11^{-1} = 8 \\ \mid \cdot 7^{-1} = 25 \\ \mid \cdot 7^{-1} = 8 \end{matrix} \tag{22}$$

$$\begin{pmatrix} 1 & 26 & 0 & 0 & 12 \\ 1 & 1 & 0 & 0 & 15 \\ 0 & 0 & 1 & 26 & 7 \\ 0 & 0 & 1 & 1 & 22 \end{pmatrix} \begin{matrix} \\ \cdot -1 \\ + \\ \\ \cdot -1 \\ + \end{matrix} \begin{pmatrix} 1 & 26 & 0 & 0 & 12 \\ 0 & 4 & 0 & 0 & 3 \\ 0 & 0 & 1 & 26 & 7 \\ 0 & 0 & 0 & 4 & 22 \end{pmatrix} \begin{matrix} \\ \mid \cdot 4^{-1} = 22 \\ \\ \mid \cdot 4^{-1} = 22 \end{matrix} \tag{23}$$

$$\begin{pmatrix} 1 & 26 & 0 & 0 & 12 \\ 0 & 1 & 0 & 0 & 8 \\ 0 & 0 & 1 & 26 & 7 \\ 0 & 0 & 0 & 1 & 11 \end{pmatrix} \begin{matrix} \\ + \\ \cdot -26 \\ \\ + \\ \cdot -26 \end{matrix} \begin{pmatrix} 7 \\ 8 \\ 11 \\ 11 \end{pmatrix} = \begin{pmatrix} H \\ I \\ L \\ L \end{pmatrix} \tag{24}$$

Bildung der Inversen K^{-1}

$$\begin{pmatrix} 7 & 8 & 1 & 0 \\ 11 & 11 & 0 & 1 \end{pmatrix} \mid \cdot 7^{-1} = 25 \quad \begin{pmatrix} 1 & 26 & 25 & 0 \\ 11 & 11 & 0 & 1 \end{pmatrix} \begin{matrix} \cdot 11^1 \\ + \end{matrix}$$

$$\begin{pmatrix} 1 & 26 & 25 & 0 \\ 0 & 15 & 15 & 1 \end{pmatrix} \mid \cdot 15^{-1} = 2 \quad \begin{matrix} + \\ \cdot 26 \end{matrix} \quad \begin{pmatrix} 1 & 0 & 28 & 6 \\ 0 & 1 & 1 & 2 \end{pmatrix}$$

$$K^{-1} = \begin{pmatrix} 28 & 6 \\ 1 & 2 \end{pmatrix} \tag{25}$$

Lösung: HILLISTEINFACHZUKNACKEN

2.6. Aufgabe 6

a. Warum gilt für zufällige Texte $I_r = 1/26 = 0.0385$?

Wirklicher Zufall würde bedeuten das jeder Buchstaben $a \in A$ gleich oft im Text vorkommt. Folglich handelt es sich um einen Laplace-Raum (wie beim Würfel) und die Warscheinlichkeit für $P(X = a) = \frac{1}{|A|}$. In unseren Fall ist $|A| = 26$.

b.

c.

2.7. Übungsaufgabe: One-Time-Pad

$$P_1 = hike = 001\ 010\ 011\ 000 \tag{26}$$
$$P_2 = rike = 101\ 010\ 011\ 000 \tag{27}$$
$$C = eier = 000\ 010\ 000\ 101 \tag{28}$$
$$K = klet = 011\ 100\ 000\ 111 \tag{29}$$
$$\tag{30}$$

a.

$$001\ 010\ 011\ 000\,\text{xor} \tag{31}$$
$$000\ 010\ 000\ 101 = \tag{32}$$
$$001\ 000\ 011\ 101 = hekr \tag{33}$$

$$101\ 010\ 011\ 000\,\text{xor} \tag{34}$$
$$000\ 010\ 000\ 101 = \tag{35}$$
$$101\ 000\ 011\ 101 = rekr \tag{36}$$

b.

$$001\ 010\ 011\ 000\,\text{xor} \tag{37}$$
$$011\ 100\ 000\ 111 = \tag{38}$$
$$010\ 110\ 011\ 111 = iskt \tag{39}$$

$$101\ 010\ 011\ 000\ \text{xor} \tag{40}$$
$$011\ 100\ 000\ 111\ = \tag{41}$$
$$110\ 110\ 011\ 010\ = sski \tag{42}$$

c.

wenn $C_{1_i} = C_{2_i} \Rightarrow P_{1_i} = P_{2_i}$

$$C_1\,\text{xor}\,C_2 = (P_1\,\text{xor}\,K)\,\text{xor}(P_2\,\text{xor}\,K) = P_1\,\text{xor}\,P_2 \tag{43}$$

d.

$$K = C_1 \operatorname{xor} P_1 \tag{44}$$

$$C_2 \operatorname{xor} K = P_2 \tag{45}$$

2.8. Skytale

a.

$$E(k, x_1, \ldots, x_{km}) = \tag{46}$$

$$x_1 x_{m+1} x_{2m+1} \ldots x_{(k-1)m+1} x_2 x_{m+2} x_{2m+2} \ldots x_{(k-1)m+2} \ldots x_m x_{2m} x_{3m} \ldots x_k m \tag{47}$$

$$\begin{pmatrix} x_1 & x_2 & \cdots & x_m \\ x_{m+1} & x_{m+2} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{(k-1)m+1} & x_{(k-1)m+2} & \cdots & x_{km} \end{pmatrix} \tag{48}$$

Ist die Klartextlänge kein Vielfaches von k, so kann der Klartext durch das Ein- bzw. Anfügen von sogenannten Blendern (Füllzeichen) verlängert werden. Damit der Emp fänger diese Füllzeichen nach der Entschlüsselung wieder entfernen kann, ist lediglich darauf zu achten, dass sie im Klartext leicht als solche erkennbar sind.

3. Doppelwürfel

3.1. Entschlüsselung des Doppelwürfels

Entschlüsseln Sie (von Hand) den folgenden Chiffretext mit den Schlüsseln K1 = *KRYPTOLOGIE* und K2 = *WISSENSCHAFT*:
ADRHIFOHCTSNIFGOETIIGHMIINTHIEIWFETÄREETPCSRSTLOSKNSDEICITEIT
ECNSBAIHMSSORDYEIE
Hinweis: Berücksichtigen Sie den unvollständigen Würfel gemäß dem Vorlesungsskript.

1. Gegeben
   $K1$ = KRYPTOLOGIE $|K1| = 11$
   $K2$ = WISSENSCHAFT $|K2| = 12$
   $|Chiffretext| = 79$
   Nächste Zahl durch 12 teilbar: $84 = 7 * 12$

2. Anordnung in Größtes Grid → Zeile

1	2	3	4	5	6	7	8	9	10	11	12
13	14	15	16	17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	×	×	×	×	×

3. Neuaufteilung, da leere Spalten hinter den ersten 7 Überlängen (79 mod 12 = 7)

W	I	S	S	E	N	S	C	H	A	F	T
A	C	E	F	H	I	N	S	S	S	T	W
1	7	13	20	26	32	39	46	53	60	67	73
2	8	14	21	27	33	40	47	54	61	68	74
3	9	15	22	28	34	41	48	55	62	69	75
4	10	16	23	29	35	42	49	56	63	70	76
5	11	17	24	30	36	43	50	57	64	71	77
6	12	18	25	31	37	44	51	58	65	72	78
×	×	19	×	×	38	45	52	59	66	×	79

4. Anordnung bei lesbarem Schlüssel

W	I	S	S	E	N	S	C	H	A	F	T
73	32	46	53	13	39	60	7	26	1	20	67
74	33	47	54	14	40	61	8	27	2	21	68
75	34	48	55	15	41	62	9	28	3	22	69
76	35	49	56	16	42	63	10	29	4	23	70
77	36	50	57	17	43	64	11	30	5	24	71
78	37	51	58	18	44	65	12	31	6	25	72
79	38	52	59	19	45	66	×	×	×	×	×

5. Anordung mit dem zweiten Schlüssel

K	R	Y	P	T	O	L	O	G	I	E
E	G	I	K	L	O	O	P	R	T	Y
73	7	47	2	41	76	10	50	5	44	79
32	26	54	21	62	35	29	57	24	65	38
46	1	14	68	9	49	4	17	71	12	52
53	20	40	75	28	56	23	43	78	31	59
13	67	61	34	3	16	70	64	37	6	19
39	74	8	48	22	42	77	11	51	25	45
60	33	27	55	69	63	36	30	58	72	66
×	×	×	15	×	×	×	×	18	×	×

6. Anordnung mit lesbarem Schlüssel

K	R	Y	P	T	O	L	O	G	I	E
2	5	79	50	44	76	41	10	7	47	73
21	24	38	57	65	35	62	29	26	54	32
68	71	52	17	12	49	9	4	1	14	46
75	78	59	43	31	56	28	23	20	40	53
34	37	19	64	6	16	3	70	67	61	13
48	51	45	11	25	42	22	77	74	8	39
55	58	66	30	72	63	69	36	33	27	60
15	18	×	×	×	×	×	×	×	×	×

7. Ersetzung der Zahlen durch die dazugehörige Chiffrenposition

D	I	E	K	R	Y	P	T	O	L	O
G	I	E	I	S	T	E	I	N	E	W
I	S	S	E	N	S	C	H	A	F	T
D	I	E	S	I	C	H	M	I	T	D
E	R	I	N	F	O	R	M	A	T	I
O	N	S	S	I	C	H	E	R	H	E
I	T	B	E	S	C	H	Ä	F	T	I
G	T	×	×	×	×	×	×	×	×	×

8. Die entschlüsselte Nachricht
   DIE KRYPTOLOGIE IST EINE WISSENSCHAFT DIE SICH MIT DER INFORMATIONSSICHERHEIT BESCHÄFTIGT

3.2. Der quadratische Doppelwürfel

In dieser Aufgabe soll der quadratische Doppelwürfel analysiert werden. Für diesen gilt: $|K| = |K1| = |K2|$ und $|Klartext| = |K|^2$.

1. Verschlüsseln Sie einen beliebigen Text mit diesem Verfahren.

2. Wobei handelt es sich bei diesem Verfahren? Worin unterscheidet sich dieses zum allgemeinen Doppelwürfel?

3. Was liegt im Fall $K1 = K2$ vor?

4. Optional: Führen Sie eine Kryptanalyse durch. Zeigen Sie eine effiziente Möglichkeit die Verschlüsselung zu brechen. Differenzieren Sie hier zwischen einem Known-Plaintext und Ciphertext-Only Angriff.

4. Moderne Symmetrische Chiffren

4.1. Lineare Abbildungen

4.2. Feistel-Cipher

a.

$F : \{0,1\}^4 \times \{0,1\}^4 \to \{0,1\}^4$ mit $F(X,Y) = X \operatorname{xor} Y$
die Rundenzahl $n = 2$,
der Plaintext $P = 10011100$ und
die Rundenschlüssel $K_1 = 0101$und $K_2 = 1100$.

Berechnen Sie den Ciphertext C.

$$L_i = R_{i-1} \tag{49}$$
$$R_i = L_{i-1} \operatorname{xor} F(R_{i-1}, K_i) \tag{50}$$

Rnd	K_i	L_i	R_i
0	—	1001	1100
1	0101	1100	0000
2	1100	0000	0000

Berechnen Sie aus C wieder den Plaintext P .

$$R_i = L_{i+1} \tag{51}$$
$$L_i = R_{i+1} \operatorname{xor} F(R_i, K_{i+1}) \tag{52}$$

Rnd	K_i	L_i	R_i
2	1100	0000	0000
1	0101	1100	0000
0	—	1001	1100

b.

Eine Feistel-Funktion ist definiert durch $F(X,Y) = X$. Berechnen Sie den Ciphertext C in Abhängigkeit von einer beliebigen Rundenzahl n und dem Plaintext $P = (L_0, R_0)$ Wie gut ist die dadurch erreichte Verschlüsselung?

Rnd	K_i	L_i	R_i
0	—	a	b
1	—	b	$b \operatorname{xor} a$
2	—	$b \operatorname{xor} a$	a
3	—	a	b

$$f(n,(a,b)) = \begin{cases} (a,b), & n \mod 3 = 0 \\ (b, b \operatorname{xor} a), & n \mod 3 = 1 \\ (b \operatorname{xor} a, a), & n \mod 3 = 2 \end{cases} \tag{53}$$

4.3. DES-Details

a. Zeigen Sie, dass die DES-Expansionspermutation eine lineare Abbildung ist.

$$A = \begin{pmatrix} 31 & 0 & 1 & 2 & 3 & 4 & 3 & 4 & 5 & 6 & 7 & 8 \\ 7 & 8 & 9 & 10 & 11 & 12 & 11 & 12 & 13 & 14 & 15 & 16 \\ 15 & 16 & 17 & 18 & 19 & 20 & 19 & 20 & 21 & 22 & 23 & 24 \\ 23 & 24 & 25 & 26 & 27 & 28 & 27 & 28 & 29 & 30 & 31 & 0 \end{pmatrix} \tag{54}$$

Sei $P \in \mathbb{N}^{32\times 48}$ die entsprechende Permutationsmatrix für A und $f : A^{32} \to A^{48}, \quad x \mapsto P \cdot x$ die Permutationsfunktion.

zZ: f ist linear

Sei $x, y \in A^{32}$ mit $x = (x_0, x_1, \ldots, x_{31})$ und $y = (y_0, y_1, \ldots, y_{31})$ dann ist

$$f(\alpha x) = \begin{pmatrix} \alpha x_{31} & \alpha x_0 & \cdots & \alpha x_8 \\ \alpha x_7 & \alpha x_8 & \cdots & \alpha x_{16} \\ \alpha x_{15} & \alpha x_{16} & \ddots & \vdots \\ \alpha x_{23} & \alpha x_{24} & \cdots & \alpha x_0 \end{pmatrix} \tag{55}$$

$$= \alpha \begin{pmatrix} x_{31} & x_0 & \cdots & x_8 \\ x_7 & x_8 & \cdots & x_{16} \\ x_{15} & x_{16} & \ddots & \vdots \\ x_{23} & x_{24} & \cdots & x_0 \end{pmatrix} \tag{56}$$

$$= \alpha f(x) \tag{57}$$

$$f(x+y) = \begin{pmatrix} x_{31}+y_{31} & x_0+y_0 & \cdots & x_8+y_8 \\ x_7+x_7 & x_8+y_8 & \cdots & x_{16}+y_{16} \\ x_{15}+y_{15} & x_{16}+y_{16} & \ddots & \vdots \\ x_{23}+y_{23} & x_{24}+y_{24} & \cdots & x_0+y_0 \end{pmatrix} \tag{58}$$

$$= \begin{pmatrix} x_{31} & x_0 & \cdots & x_8 \\ x_7 & x_8 & \cdots & x_{16} \\ x_{15} & x_{16} & \ddots & \vdots \\ x_{23} & x_{24} & \cdots & x_0 \end{pmatrix} + \begin{pmatrix} y_{31} & y_0 & \cdots & y_8 \\ y_7 & y_8 & \cdots & y_{16} \\ y_{15} & y_{16} & \ddots & \vdots \\ y_{23} & y_{24} & \cdots & y_0 \end{pmatrix} \tag{59}$$

$$= f(x) + f(y) \tag{60}$$

#

b. Zeigen Sie, dass die DES S-Boxen keine lineare Abbildung sind.

zZ. S_i **ist nicht linear.** Wir nehmen die S1-Box und sei $x = 0 \wedge \alpha = 2$

$$\begin{aligned} \mathrm{S1}(\alpha x) &= \mathrm{S1}(000000_2) && (61)\\ &= 1110_2 = 14_{10} && (62) \end{aligned}$$

$$\begin{aligned} \alpha\,\mathrm{S1}(x) &= 2\,\mathrm{S1}(000000_2) && (63)\\ &= 2_{10} * 1110_2 = 28_{10} && (64) \end{aligned}$$

$$\Rightarrow \mathrm{S1}(\alpha x) \neq \alpha\,\mathrm{S1}(x) \quad (65)$$

#

c. Geben Sie für die DES-Permutation die Zykelschreibweise an.

$$A = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 \\ 15 & 6 & 19 & 20 & 28 & 11 & 27 & 16 & 0 & 14 & 22 & 25 & 4 & 17 & 30 & 9 & 1 & 7 & 23 & 13 & 31 & 26 & 2 \end{pmatrix} \quad (66)$$

Zyklen:(0 15 9 14 30 3 20 31 24 18 23 8)(1 6 27 5 11 25 12 4 28 21 26 29 10 22 2 19 13 17 7 16) (67)

d. Zeigen Sie, dass für den DES Schlüssel $K = 0xE0E0E0E0F1F1F1F1$ **(inklusive Parity-Bits) alle Rundenschlüssel identisch sind. Warum gilt in diesem Fall** $DES(K, DES(K, M)) = M$ **(d.h. Ver- und Entschlüsselung sind identisch)?**

$$K = \begin{matrix} 1&1&1&0&0&0&0&0 \\ 1&1&1&0&0&0&0&0 \\ 1&1&1&0&0&0&0&0 \\ 1&1&1&0&0&0&0&0 \\ 1&1&1&1&0&0&0&1 \\ 1&1&1&1&0&0&0&1 \\ 1&1&1&1&0&0&0&1 \\ 1&1&1&1&0&0&0&1 \end{matrix} \xrightarrow{PC1} \begin{matrix} 1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1 \\ 0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0 \end{matrix} \quad (68)$$

Da alle in den folgenden Runden lediglich zu Spaltentransposition kommt und jede Spalte gleich sind, sind alle C_i, D_i für alle $1 \leq i \leq 16$.

4.4. Meet-in-the-Middle-Angriff auf 3DES

Die Erklärungen basieren auf einem known-plain/ciphertext Angriff und sollen die benutzten Schlüsselpaare ergeben.

Legende:

$$P = Plaintext$$
$$C = Ciphertext$$
$$K_x = Key_x$$

a. Angriff auf DES2

- Verschlüsselung von DES2:

$$D(K_2, E(K_1, P)) = C \tag{69}$$

- Aufteilung der Verschlüsselung in zwei Funktionen:

$$\overbrace{E(K_2, C)}^{A} = C_{middle} = \overbrace{E(K_1, P)}^{B} \tag{70}$$

- Mögliche Schlüsselmöglichkeiten:

$$A = 2^{56}$$
$$B = 2^{56}$$
$$A + B = 2^{56} + 2^{56}$$
$$= 2 * 2^{56}$$
$$= 2^{57}$$

4.5. Rechnen in $GF(2^3)$

a. Berechnen Sie das Produkt der Elemente 5 * 3 sowie die Summe der Elemente 5 + 3 für das Modularpolynom $M(x) = x3 + x + 1$.

$$5 = 101\ ,\ 3 = 011$$

Summe: $101 \operatorname{xor} 011 = 110 = 6$

Produkt:

$$101 \cdot 011 = 1111 \tag{71}$$
$$1111 \operatorname{xor} 1011 = 100 = 4 \tag{72}$$

b. Der erweiterte Euklidische Algorithmus kann benutzt werden, um auch für Elemente in einem Erweiterungskörper $GF(2^r)$ multiplikativ inverse Elemente zu berechnen. Das Modularpolynom sei $M(x) = \{100011011\}$ vom Grad $r = 8$. Dieses Modularpolynom wird auch von AES benutzt. Berechnen Sie zu $a(x) = \{00001101\}$ das multiplikativ inverse Element $a^{-1}(x)$.

$$M(x) = 111000 \cdot a(x) + 11$$
$$a(x) = 100 \cdot 11 + 1$$

$$1 = a(x) + 100 \cdot 11 \tag{73}$$
$$= a(x) + 100 \cdot (M(x) + 111000 \cdot a(x)) \tag{74}$$
$$= a(x) + 100 \cdot M(x) + 111000 \cdot 100 \cdot a(x) \tag{75}$$
$$= a(x) + 11100000 \cdot a(x) \tag{76}$$
$$= 11100001 \cdot a(x) \tag{77}$$
$$\rightarrow a^{-1}(x) = 11100001 = x^7 + x^6 + x^5 + 1 = 225 \tag{78}$$

Nebenrechnungen:
1.Reihe

```
100011011
1101
 1011
 1101
  1101
  1101
     000
```

$$r = 11 \quad q = 111000$$

$$110100000 + 11010000 + 1101000 = 1101 * (x^5 + x^4 + x^3) = 1101 * 111000$$

2.Reihe

```
1101
11
```

$$r = 1 \quad q = 100$$

4.6. $GF(2^3)$

a. Finden Sie alle irreduziblen Polynome vom Grad 3 mit Koeffizienten aus $GF(2)$.

	010	011	100	110	101	111
010	100	110	1000	1100	1010	1110
011		101	1100	1010	1111	1001
100			10000	$\cdots$	$\cdots$	$\cdots$
110				$\ddots$	$\cdots$	$\vdots$
101					$\ddots$	$\vdots$
111						10101

reduzible sind: 1001, 1010, 1100, 1110, 1111, =9,10,12,14,15

b. Definieren Sie den Körper mit 8 Elementen, indem Sie ein passendes irreduzibles Polynom und die Verknüpfungstabellen für Addition und Multiplikation angeben.

4.7. AES-MixColumns-Beispiel

a. Berechnen Sie eine Spaltentransformation durch MixColumns

- Gegeben: $\begin{pmatrix} d4 \\ bf \\ 4d \\ 30 \end{pmatrix}$
- INT to HEX:

INT	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HEX	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

- HEX to BYTE/BIT

HEX		$Byte_1(INT)$	$Byte_0(INT)$		$Byte_1$	$Byte_0$		State
d4	$\rightarrow$	13	4	$\rightarrow$	1101	0100	=	S_0
bf	$\rightarrow$	11	15	$\rightarrow$	1011	1111	=	S_1
4d	$\rightarrow$	4	13	$\rightarrow$	0100	1101	=	S_2
30	$\rightarrow$	3	0	$\rightarrow$	0011	0000	=	S_3

- Definiton der MixColumns-Operation:

$$\begin{bmatrix} S'_{0,C} \\ S'_{1,C} \\ S'_{2,C} \\ S'_{3,C} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} * \begin{bmatrix} S_{0,C} \\ S_{1,C} \\ S_{2,C} \\ S_{3,C} \end{bmatrix} \tag{79}$$

- MixColumns-Operation:

$$\begin{bmatrix} S'_{0,C} \\ S'_{1,C} \\ S'_{2,C} \\ S'_{3,C} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} * \begin{bmatrix} d4 \\ bf \\ 4d \\ 30 \end{bmatrix} \tag{80}$$

- $S'_{0,C}$:

$$S'_{0,C} = (\{02\} * S_{0,C}) \oplus (\{03\} * S_{1,C}) \oplus S_{2,C} \oplus S_{3,C} \tag{81}$$

  - $(\{02\} * S_{0,C})$

0	0	0	0	0	0	1	0	*	1	1	0	1	0	1	0	0
						1	0									
							1	0								
								0	0							
									1	0						
										0	0					
											1	0				
												0	0			
													0	0		
						1	1	0	1	0	1	0	0	0		
					⊕	1	0	0	0	1	1	0	1	1		
						0	1	0	1	1	0	0	1	1		
							1	0	1	1	0	0	1	1	=	B3

– $(\{03\} * S_{1,C})$

0	0	0	0	0	0	1	1	*	1	0	1	1	1	1	1	1
						1	1									
							0	0								
								1	1							
									1	1						
										1	1					
											1	1				
												1	1			
													1	1		
						1	1	1	0	0	0	0	0	1		
					⊕	1	0	0	0	1	1	0	1	1		
						0	1	1	0	1	1	0	1	0		
							1	1	0	1	1	0	1	0	=	DA

– $\{B3\} \oplus \{DA\} \oplus \{4C\} \oplus \{30\}$

	1	0	1	1	0	0	1	1	
	1	1	0	1	1	0	1	0	
	0	1	0	0	1	1	0	1	
⊕	0	0	1	1	0	0	0	0	
	0	0	0	1	0	1	0	0	
				1	0	1	0	0	= {14}

- $S'_{1,C}$:

$$S'_{1,C} = S_{0,C} \oplus (\{02\} * S_{1,C}) \oplus (\{03\} * S_{2,C}) \oplus S_{3,C} \tag{82}$$

– $(\{02\} * S_{1,C})$

0	0	0	0	0	0	1	0	*	1	0	1	1	1	1	1	1
						1	0									
							0	0								
								1	0							
									1	0						
										1	0					
											1	0				
												1	0			
													1	0		
						1	0	1	1	1	1	1	1	0		

– $(\{03\} * S_{2,C})$

0	0	0	0	0	0	1	1	*	0	1	0	0	1	1	0	1
						0	0									
							1	1								
								0	0							
									0	0						
										1	1					
											1	1				
												0	0			
													1	1		
							1	1	0	1	0	1	1	1		

- $\{D4\} \oplus \{xx\} \oplus \{xx\} \oplus \{30\}$

		1	1	0	1	0	1	0	0		
	1	0	1	1	1	1	1	1	0		
		1	1	0	1	0	1	1	1		
$\oplus$		0	0	1	1	0	0	0	0		
	1	0	1	0	0	1	1	0	1		
$\oplus$	1	0	0	0	1	1	0	1	1		
	0	0	1	0	1	0	1	1	0		
		0	1	0	1	0	1	1	0	=	$\{56\}$

- $S'_{2,C}$:

$$S'_{2,C} = S_{0,C} \oplus S_{1,C} \oplus (\{02\} * S_{2,C}) \oplus (\{03\} * S_{3,C}) \tag{83}$$

– $(\{02\} * S_{2,C})$

0	0	0	0	0	0	1	0	*	0	1	0	0	1	1	0	1
						0	0									
							1	0								
								0	0							
									0	0						
										1	0					
											1	0				
												0	0			
													1	0		
						0	1	0	0	1	1	0	1	0		

– $(\{03\} * S_{3,C})$

0	0	0	0	0	0	1	1	*	0	0	1	1	0	0	0	0
						0	0									
							0	0								
								1	1							
									1	1						
										0	0					
											0	0				
												0	0			
													0	0		
								1	0	1	0	0	0	0		

– $\{D4\} \oplus \{BF\} \oplus \{xx\} \oplus \{xx\}$

		1	1	0	1	0	1	0	0		
		1	0	1	1	1	1	1	1		
		1	0	0	1	1	0	1	0		
	$\oplus$	0	1	0	1	0	0	0	0		
		1	0	1	0	0	0	0	1	=	$\{A1\}$

- $S'_{3,C}$:

$$S'_{3,C} = (\{03\} * S_{0,C}) \oplus S_{1,C} \oplus S_{2,C} \oplus (\{02\} * S_{3,C}) \tag{84}$$

– $(\{03\} * S_{0,C})$

0	0	0	0	0	0	1	1	*	1	1	0	1	0	1	0	0
						1	1									
							1	1								
								0	0							
									1	1						
										0	0					
											1	1				
												0	0			
													0	0		
						1	0	1	1	1	1	1	0	0		

– $(\{02\} * S_{3,C})$

0	0	0	0	0	0	1	0	*	0	0	1	1	0	0	0	0
						0	0									
							0	0								
								1	0							
									1	0						
										0	0					
											0	0				
												0	0			
													0	0		
								1	1	0	0	0	0	0		

- $\{xx\} \oplus \{BF\} \oplus \{4D\} \oplus \{xx\}$

	1	0	1	1	1	1	1	0	0		
			1	1	0	0	0	0	0		
		1	0	1	1	1	1	1	1		
$\oplus$		0	1	0	0	1	1	0	1		
	1	1	1	1	0	1	1	1	0		
$\oplus$	1	0	0	0	1	1	0	1	1		
	0	1	1	1	1	0	1	0	1		
		1	1	1	1	0	1	0	1	=	$\{F5\}$

- Ergebnis:

$$\begin{bmatrix} 14 \\ 56 \\ A1 \\ F5 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} * \begin{bmatrix} d4 \\ bf \\ 4d \\ 30 \end{bmatrix} \tag{85}$$

b. Überprüfen Sie das Ergebnis, indem das Ergebnis mit der Inversen Matrix multiplizieren und wieder die Ausgangsspalte erhalten.

Hinweis: Die inverse Matrix finden Sie in FIPS 197 (S.27 InvMixColumns()) .

- Definition der inversen MixColumns-Operation:

$$\begin{bmatrix} S'_{0,C} \\ S'_{1,C} \\ S'_{2,C} \\ S'_{3,C} \end{bmatrix} = \begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} * \begin{bmatrix} S_{0,C} \\ S_{1,C} \\ S_{2,C} \\ S_{3,C} \end{bmatrix} \tag{86}$$

- Berechnung durch die inverse Matrix zur Lösungsprobe:
  Siehe mixColumnsInv() in Java. Danach stimmt die Lösung.

c. MixColumns in Java

- MixColumns()

Listing 1: mixColumns

```
public int[] mixColumns( int[] array )
{
    int[] res = new int[array.length];

    //S'0C
    //({02} * S0) XOR ({03} * S1) XOR S2 XOR S3
    int s0 = galoaMult( 2, array[0] )
             ^ galoaMult( 3, array[1] )
             ^ array[2] ^ array[3];
    res[0] = s0;

    //S'1C
    //S0 XOR ({02} * S1) XOR ({03} * S2) XOR S3
    int s1 = array[0] ^ galoaMult( 2, array[1] )
             ^ galoaMult( 3, array[2] ) ^ array[3];
    res[1] = s1;

    //S'2C
    //S0 XOR S1 XOR ({02} * S2) XOR ({03} * S3)
    int s2 = array[0] ^ array[1]
             ^ galoaMult( 2, array[2] )
             ^ galoaMult( 3, array[3] );
    res[2] = s2;

    //S'3C
    //({03} * S0) XOR S1 XOR S2 XOR ({02} * S3)
    int s3 = galoaMult( 3, array[0] ) ^ array[1]
             ^ array[2] ^ galoaMult( 2, array[3] );
    res[3] = s3;

    return res;
}//mixColumns()
```

- Multiplikation im GF

Listing 2: galoaMult

```
public int galoaMult( int a, int b )
{
    int res = 0;
    int t;
```

```
        byte aa = (byte) a;
        byte bb = (byte) b;

        while( aa != 0 )
        {
            if( ( aa & 1 ) != 0 )
            {
                res = (byte) ( res ^ bb );
            }//if
            t = (byte) ( bb & 0x80 );
            bb = (byte) ( bb << 1 );
            if( t != 0 )
            {
                bb = (byte) ( bb ^ 0x1b );
            }//if

            aa = (byte) ( ( aa & 0xff ) >> 1 );
        }//while

        if( res < 0 )
        {
            res += 256;
        }//if

        return res;
    }//galoaMult()
```

d. MixColumnsInverse in Java

- MixColumnsInv()

Listing 3: mixColumnsInv

```
    public int[] mixColumnsInv( int[] array )
    {
        int[] res = new int[array.length];

        //S'0C
        //({14} * S0) XOR ({11} * S1) XOR ({13} * S2) XOR ({09} * S3
        int s0 = galoaMult( 14, array[0] ) ^ galoaMult( 11, array[1]
                    ^ galoaMult( 13, array[2] )
                    ^ galoaMult( 9, array[3] );
        res[0] = s0;

        //S'1C
```

```
        //({09} * S0) XOR ({14} * S1) XOR ({11} * S2) XOR ({13} * S3
        int s1 = galoaMult( 9, array[0] ) ^ galoaMult( 14, array[1]
                    ^ galoaMult( 11, array[2] )
                    ^ galoaMult( 13, array[3] );
        res[1] = s1;

        //S'2C
        //({13} * S0) XOR ({09} * S1) XOR ({14} * S2) XOR ({11} * S3
        int s2 = galoaMult( 13, array[0] ) ^ galoaMult( 9, array[1]
                    ^ galoaMult( 14, array[2] )
                    ^ galoaMult( 11, array[3] );
        res[2] = s2;

        //S'3C
        //({11} * S0) XOR ({13} * S1) XOR ({09} * S2) XOR ({14} * S3
        int s3 = galoaMult( 11, array[0] ) ^ galoaMult( 13, array[1]
                    ^ galoaMult( 9, array[2] )
                    ^ galoaMult( 14, array[3] );
        res[3] = s3;

        return res;
    }//mixColumnsInv()
```

- Multiplikation im GF:
  siehe Multiplikation in GF unter MixColumns in Java

4.8. A5/1

$$X = (x_0, x_1 \cdots, x_{18}) = (1010101010101010101) \tag{87}$$
$$Y = (y_0, y_1, \cdots, y_{21}) = (1100110011001100110011) \tag{88}$$
$$Z = (z_0, z_1, \cdots, z_{22}) = (11100001111000011110000) \tag{89}$$

while *True* **do**
  $m = maj(x_8, y_{10}, z_{10})$;
  if $x_8 = m$ **then**
    $t = x13 \text{ xor } x16 \text{ xor } x17 \text{ xor } x18$;
    shift t into X;
  if $y_{10} = m$ **then**
    $t = y_{20} \text{ xor } y_{21}$;
    shift t into Y;
  if $z_{10} = m$ **then**
    $t = z_7 \text{ xor } z_{20} \text{ xor } z_{21} \text{ xor } z_{22}$;
    shift t into Y;
  $k_i = x_{18} \text{ xor } y_{21} \text{ xor } z_{22}$;

a.

Rnd	x_8	y_{10}	z_{10}	m	X'	Y'	Z'	Bit
1	1	0	1	1	0101010101010101010*1*	11001100110011001100110011	011100001111000011110000*0*	1
2	0	0	1	0	0010101010101010101*0*	01100110011001100110011*1*	01110000111100001111000	0
3	1	1	1	1	1001010101010101010*1*	00110011001100110011001*1*	10111000011110000111100*0*	0

b.

```
package ueb4;

public class A51 {
    ShiftRegister x, y, z;

    public A51() {
        x = new ShiftRegister(19,  349525);
        y = new ShiftRegister(22, 3355443);
        z = new ShiftRegister(23, 7401712);
    }

    public int maj(int i, int j, int k) {
        if ((i == 0 && j == 0) || (j == 0 && k == 0) || (i == 0 && k == 0))
            return 0;
        return 1;
    }

    public int next() {
        System.out.format("\t%d %d %d%n", x.get(8), y.get(10), z.get(10));
        int m = maj(x.get(8), y.get(10), z.get(10));
```

```
        System.out.println("M:" + m);

        if (m == x.get(8)) {
            int t = x.get(13) ^ x.get(16) ^ x.get(17) ^ x.get(18);
            x.push(t);
        }
        if (m == y.get(10)) {
            int t = y.get(20) ^ y.get(21);
            y.push(t);
        }

        if (m == z.get(10)) {
            int t = z.get(7) ^ z.get(20) ^ x.get(21) ^ x.get(22);
            z.push(t);
        }
        return (x.get(18) ^ y.get(21) ^ z.get(22));
    }

    @Override
    public String toString() {
        return "x: " + x + "\ny: " + y + "\nz: " + z;
    }

    public static void main(String[] args) {
        A51 c = new A51();

        System.out.println(c);

        for (int i = 0; i < 3; i++) {
            System.out.println(c.next());
            System.out.println(c);
        }
        System.out.println();
    }

    private static void checkBin(int i) {
        if (i != 0 && i != 1)
            throw new IllegalArgumentException();
    }

    class ShiftRegister {
        int reg;
        final int size;
```

```
        public ShiftRegister(int size, int content) {
            if (size >= 64)
                throw new IllegalArgumentException();
            this.size = size;
            reg = content;
        }

        public void push(int t) {
            checkBin(t);
            reg = (reg << 1) + t;
        }

        public int get(int i) {
            return (int) ((reg >> i) & 1);
        }

        @Override
        public String toString() {
            StringBuilder b = new StringBuilder();
            for (int i = 0; i < size; i++) {
                b.append(get(i));
            }
            return b.toString();
        }
    }
}
```

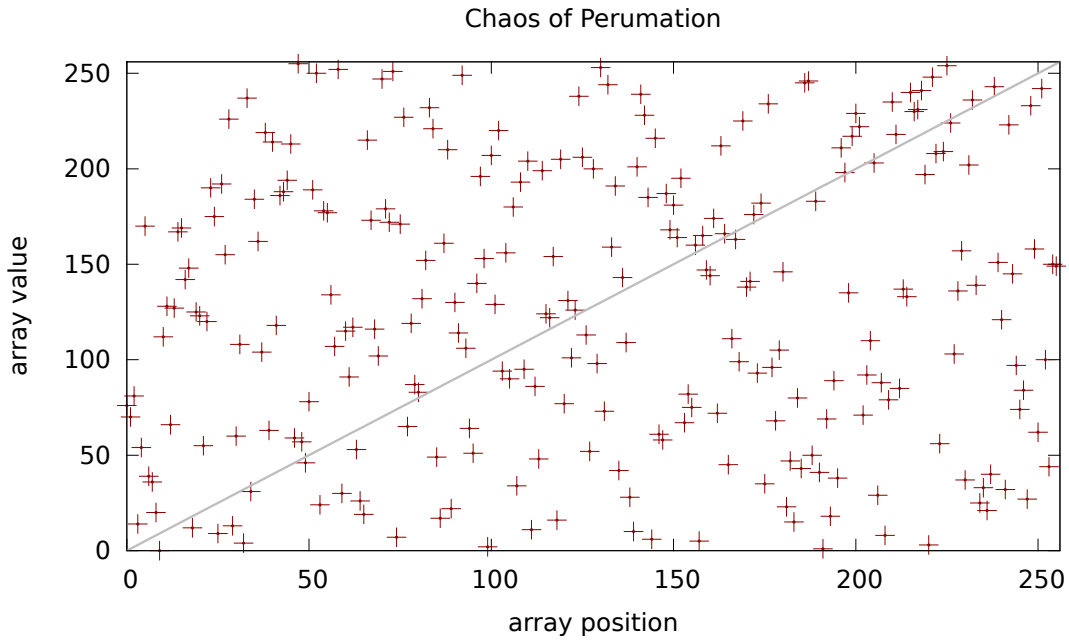
4.9. RC4

```
package ueb4;

import java.util.Arrays;

import com.panayotis.gnuplot.*;
import com.panayotis.gnuplot.plot.*;
import com.panayotis.gnuplot.style.*;
import com.panayotis.gnuplot.terminal.*;

public class RC4 {
    int[] k, s = new int[256];
    byte L;

    public RC4(int[] k2) {
        this.k = k2;
```

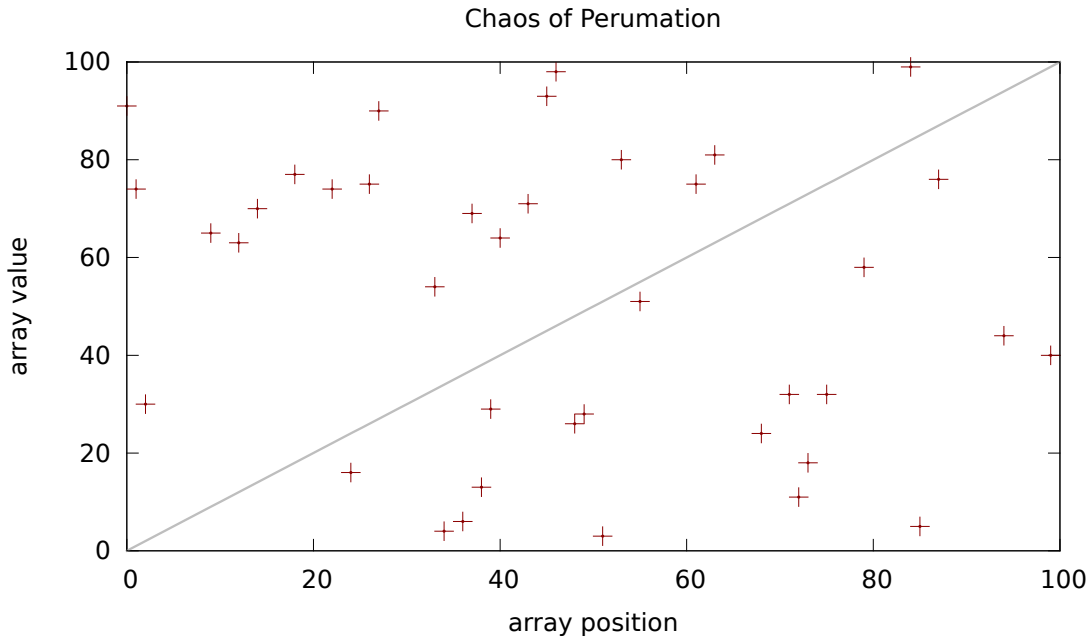
```
        L = (byte) k2.length;
        init();
    }

    private void init() {
        for (int i = 0; i < s.length; i++) {
            s[i] = (i % 256);
            assert s[i] >= 0;
        }
        int j = 0;
        for (int i = 0; i < s.length; i++) {
            j = (j + s[i] + k[i % L]) % 256;
            swap(s, j, i);
        }
    }

    private void swap(int[] b, int j, int i) {
        int t = b[i] % 256;
        b[i] = b[j];
        b[j] = t;
    }

    public void next(int[] ciph) {
        int i = 0, j = 0;
        for (int n = 0; n < ciph.length; n++) {
            i = (i + 1) % 256;
            j = (j + s[i]) % 256;
            swap(s, i, j);
            int rand = s[(s[i] + s[j]) % 256];
            ciph[n] = rand;
        }
    }

    @Override
    public String toString() {
        return Arrays.toString(s);
    }

    private static void plot(String file, int k[]) {
        JavaPlot plot = new JavaPlot();
        double[][] data = new double[k.length][2];
        for (int i = 0; i < k.length; i++) {
            data[i][0] = i;
            data[i][1] = k[i];
```

```
        }

        plot.setTitle("Chaos of Perumation");
        plot.setTerminal(new FileTerminal("pdf", file));

        DataSetPlot dataP = new DataSetPlot(data);
        PlotStyle ps = new PlotStyle(Style.POINTS);
        ps.setLineType(NamedPlotColor.DARK_RED);
        dataP.setPlotStyle(ps);
        dataP.setTitle("");
        plot.addPlot(dataP);

        FunctionPlot dataL = new FunctionPlot("x");
        PlotStyle rp = new PlotStyle(Style.LINES);
        rp.setLineType(NamedPlotColor.GRAY);
        rp.setLineWidth(3);
        dataL.setPlotStyle(rp);
        dataL.setTitle("");
        plot.addPlot(dataL);

        plot.getAxis("x").setLabel("array position");
        plot.getAxis("x").setBoundaries(0, k.length);
        plot.getAxis("y").setLabel("array value");
        plot.getAxis("y").setBoundaries(0, k.length);

        // GNUPlot.getDebugger().setLevel(Debug.VERBOSE);
        plot.plot();
    }

    public static void main(String[] args) {
        int k[] = { 0x1A, 0x2B, 0x3C, 0x4D, 0x5E, 0x6F, 0x77 };
        RC4 rc4 = new RC4(k);
         System.out.println("S:"+rc4);
//      plot("beforerc4.pdf", rc4.s);
        int b[] = new int[100];
        rc4.next(b);
        plot("rc4-key-seq.pdf", b);
         System.out.println("S:" +rc4);
//      plot("afterrc4.pdf", rc4.s);
        System.out.println(Arrays.toString(b));
    }
}
```

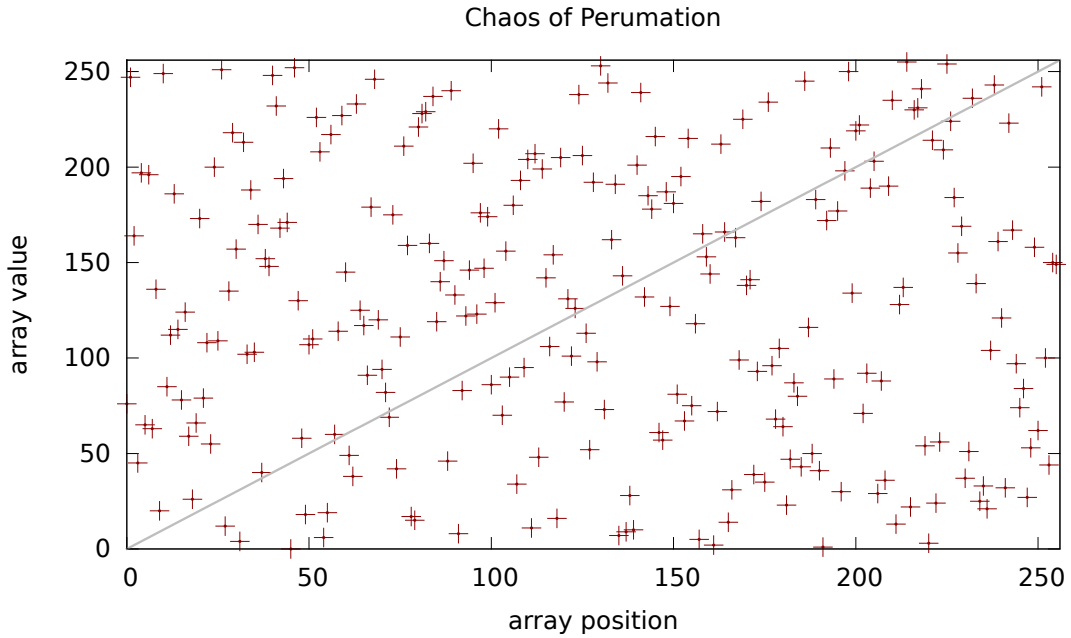
a. Listen Sie die Permuation S nach der Initialisierung auf.



b. Generieren Sie 100 Schlüsselbytes.



c. Listen Sie die Permuation S erneut auf.



5. Hashfunktionen und MACs

5.1. Funktionsweise von Hash-Funktionen

$$Y = 11110 \tag{90}$$
$$F: \{0,1\}^4 \to \{0,1\}^2, \tag{91}$$
$$(x_1, x_2, x_3, x_4) \mapsto (x_1 \text{ xor } x_4, x_2 \text{ xor } x_3) \tag{92}$$

x_1	x_2	x_3	x_4	$F(x_1, x_2, x_3, x_4)$
1	0	1	1	10
1	0	1	1	10
0	1	0	0	01

5.2. Kollisionen und Preimage-Angriffe

$$X = (X_0, X_1, X_2, \ldots, X_{n-1})$$

$$H(X) = nX_0 + (n-1)X_1 + (n-2)X_2 + \ldots + 2X_{n-2} + X_{n-1} \mod 256 = \sum_{i \geq 0}^{n} (n-i)X_i \mod 256$$

a. Berechnen Sie H(X) für die Nachricht "FHT4ever". Interpretieren Sie dabei jeden Buchstaben als seinen 8 Bit ASCII-Wert.

$$X = (70, 72, 84, 52, 101, 118, 101, 114) \tag{93}$$
$$H(X) = 8 \cdot 70 + 7 \cdot 72 + 6 \cdot 84 + 5 \cdot 52 + 4 \cdot 101 + 3 \cdot 118 + 2 \cdot 101 + 114 \tag{94}$$
$$= 560 + 504 + 504 + 260 + 404 + 354 + 202 + 114 \tag{95}$$
$$= 48 + 248 + 248 + 4 + 148 + 98 + 202 + 114 \tag{96}$$
$$= 86 \tag{97}$$

```
H = lambda x: sum([(len(x)-i)*ord(e)\%256 for i,e in enumerate(x)])\%256
```

b. Finden Sie eine andere (sinnvolle) Nachricht, die den gleichen Hashwert wie "FHT4ever" hat.

$$H(Uni4ever) = 86$$

c. Gegeben sei $h(X) = 42$, wobei $X = (X_0, X_1, X_2)$. Finden Sie ein $Y = (Y_0, Y_1, Y_2)$ mit $X \neq Y \wedge h(Y) = h(X)$.

d. Finden Sie eine weitere Kollision.

$$H(X) = 42 = (3X_1 + 2X_2 + X_1) \mod 256 \tag{98}$$

Folgende Tabelle gibt Tupeln $X = (X_0, X_1, X_2)$ mit $H(X) = 42$ an. Insgesamt existieren 65536 Kollision (ExhaustedSearch).

```
c = [ (x,y,z) for x in range(256)
              for y in range(256)
                for z in range(256)
                  if (3*x+2*y+z) % 256==42 ]
print len(c)
```

X_1	X_2	X_3
96	166	190
96	167	188
96	168	186

5.3. Das Online-Auktionshaus

a. Denken Sie sich ein möglichst einfaches Verfahren aus, das auf einer Hash-Funktion beruht.

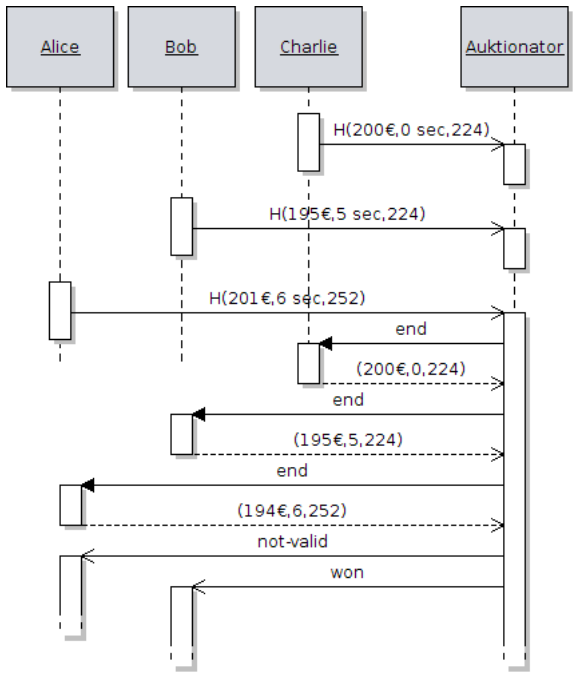
Ein Gebot ist ein Hashwert bestehend aus: Betrage, Timestamp und Nonce

$$H((bid \circ time \circ salt)) = Y$$

Die anderen Bieter können nicht errechnen welchen Betrag ein Bieter geboten hat, da die Umkehrung von H nicht möglich ist.

Sobald die Bieterrunde geschlossen wird, schickt jeder Bieter sein Gebot, Timestamp und Nonce zum Auktionator. Dieser kann nun die Hashwerte der Gebote überprüfen und den günstigen Bieter ermitteln.

Folgendes Bild zeigt ein Beispielablauf einer Auktion.



Im Beispiel sehen, dass Alices Gebot abgelehnt wird da

$$H(201 \circ \ldots) = H(196 \circ \ldots)$$

Sowie das Bob gewinnt, da er das günstigste Gebot abgeben hat.

b. Welche Angriffe gibt es trotzdem noch auf das Verfahren?

- Man-in-the-Middle Attake aufgrund fehlender Authentizität.
- Störungen des Übertragungskanals
- ohne Timetamp wäre Replayangriffe möglich
- ohne Salt wäre die Authentifizierung komplett ausgehebelt
- Brute-Force-Attake ausprobieren von Beträgen,...
- Zeit muss synkron sein.

Wambach: $H(Name \circ Betrag)$ kann jeder berechnen.
Münze werfen Internet: X,Y wählen eine große Zahl. Austausch der Hashwert der Zahlen. X,Y austauschen. X+Y ist die Zufallszahl.

5.4. Datenbankschutz durch Verschlüsselung und Hash-Funktionen

```
-- Alexander Weigl <weigla@fh-trier.de>
-- Date: 2011-05-02
-- Simple membership management with crypto functions on mysql
-- call: mysql -u <user> -p<password> < member.db.sql
--       for creation

drop database if exists krypt;
create database krypt;
use             krypt;

CREATE TABLE members (
  idx varchar(40)   not null primary key comment 'sha 1 key of lastname',
  data varchar(1000) not null unique     comment 'crypted data from person'
);

DELIMITER //

CREATE FUNCTION getMember (lastname varchar(40) )
RETURNS varchar(256)
BEGIN
```

```
        DECLARE tmp VARCHAR(1000);
        SELECT data FROM members
        WHERE  idx = SHA1(lastname) INTO tmp;
        RETURN AES_DECRYPT(tmp,lastname);
END;//

CREATE PROCEDURE saveMember(IN lastname varchar(40)
                          , IN data varchar(256)    )
BEGIN
        DECLARE k VARCHAR(40);
        DECLARE v VARCHAR(1000);

        SET v= AES_ENCRYPT(data, lastname);
        SET k= SHA1(lastname);

        INSERT INTO members VALUES (k, v);
END;//

delimiter ;

CALL saveMember("Weigl", "Alexander Weigl -- Hornstr 11 -- 54294 Trier");
CALL saveMember("Wambach", "Tim Wambach -- Somewhere ...");
CALL saveMember("Kuenkler", "Andreas Kuenkler -- Somewhere ...");
CALL saveMember("Knor", "Konstantin Knor -- Somewhere ...");
CALL saveMember("Yuen", "Timmy Wai Hong Yuen -- Am Bahnhof, im schlimmen Vier


SELECT * FROM members;

SELECT getMember("Weigl");
SELECT getMember("Yuen");
```

5.5. kryptologische Absicherung der Prüfungsvorleistung

a. Das Verfahren ist bisher kryptologisch nicht gesichert. Welche Angriffe sind denkbar?

- Identitätsdiebstahl, man verwendet den Zettel eines anderen
- Replikation des eigenen Scheines mit Fälschung des Ergebnisses
- Replikation eines anderen Scheines Fälschung der persönlichen Angaben

b. Sichern Sie das Verfahren kryptologisch ab. Der Professor soll die "Echtheit" der Bescheinigung möglichst einfach prüfen können.

(1) Auf dem Schein wird ein QR-Code abgedruckt der einen Hash mit den Angaben auf dem Schein und einen geheimen Saltwert beinhaltet. Dies kann mit einem Handy leicht geprüft. Überprüfung der Identität ist weiterhin erforderlich.

(2) Verfahren (1) kann auch als Hex-Zeichen aufgedruckt werden.

(3) $H(Matrikel, Bestanden, Salt) = (Matrikel + Bestanden + Salt \mod N)$

6. Übungsaufgaben: Asymmetrische Kryptologie

6.1. Rucksack

PrivateKey: $(3, 5, 10, 23), m = 8, n = 47$

a. Geben sie den öffentlichen Schlüssel an

$$3 \cdot 8 \mod 47 = 24 \tag{99}$$
$$5 \cdot 8 \mod 47 = 40 \tag{100}$$
$$10 \cdot 8 \mod 47 = 33 \tag{101}$$
$$23 \cdot 8 \mod 47 = 43 \tag{102}$$

PublicKey: $(24, 40, 33, 43)$

b. Verschlüsseln Sie $P = [111000000010]_2$ (Binärdarstellung) im ECB-Modus.

$$C_1 = 1 \cdot 24 + 1 \cdot 40 + 1 \cdot 33 + 0 \cdot 43 = 97 \tag{103}$$
$$C_2 = 0 \cdot 24 + 0 \cdot 40 + 0 \cdot 33 + 0 \cdot 43 = 0 \tag{104}$$
$$C_3 = 0 \cdot 24 + 0 \cdot 40 + 0 \cdot 33 + 0 \cdot 43 = 33 \tag{105}$$

c. Finden Sie den Plaintext zum Ciphertext $C = (67, 64)$

$m^{-1} = 6$:

$C_1 = 67 * 6 \mod 47 = 26$

S_i	Cm^{-1}	P
23	26	1
10	3	0
5	3	0
3	0	1

$P_1 = 1001$

$C_2 = 64 * 6 \mod 47 = 8$

S_i	Cm^{-1}	P
23	8	0
10	8	0
5	3	1
3	0	1

$P_2 = 0011$

6.2. RSA auf Nachricht in Blöcken

$$P = \text{'FHT4EVER'} \qquad n = 13 \cdot 17 = 221 \qquad e = 3 \tag{106}$$

Beachten: $ggT(e, \phi(221)) \neq 1$. Entschlüsselung damit ummöligch.

$$E(x) = x^3 \mod 221 \tag{107}$$
$$E(P) = E('F') \circ E('H') \circ E('T') \circ E('4') \circ E('E') \circ E('V') \circ E('E') \circ E('R') \tag{108}$$
$$= \text{apply}(x^3 \mod 221, [70, 72, 84, 52, 69, 86, 69, 82]) \tag{109}$$
$$= [8, 200, 203, 52, 103, 18, 103, 194] \tag{110}$$

6.3. Chinesischer Restsatz

a. Es sei $m = 11, n = 12, a = 3$ **und** $b = 4$**. Geben Sie ein** x **an, für das gilt:** $x \mod m = a$ **und** $x \mod n = b$

$$x = anN + bmM = 3 \cdot 12 \cdot N + 4 \cdot 11M \tag{111}$$

$$ggT(12, 11) = 1 \text{ mit } a^{-1} = -1 = N$$

$$ggT(11, 12) = 1 \text{ mit } a^{-1} = 1 = M$$

$$X = 3 \cdot 12 \cdot 1 + 4 \cdot 11 \cdot -1 \tag{112}$$
$$= 36 - 44 = -8 \tag{113}$$

Probe:

$$-8 \mod m = a \qquad -8 \mod n = b \tag{114}$$
$$-8 \mod 11 = 3 \qquad -8 \mod 12 = 4 \tag{115}$$
$$\tag{116}$$

b. Es sei $m = 11, n = 12, l = 12, a = 3, b = 4$ **und** $c = 5$**. Geben Sie ein** x **an, für das gilt:** $x \mod m = a$ **und** $x \mod n = b$ **und** $x \mod l = c$**.**

$$X = a \cdot n \cdot l(nl)^{-1} + b \cdot m \cdot l(ml)^{-1} + c \cdot n \cdot m(nm)^{-1} \tag{117}$$
$$x \mod 11 = 3 \tag{118}$$
$$x \mod 12 = 4 \tag{119}$$
$$x \mod 11 = 5 \tag{120}$$
$$\tag{121}$$

Vorrausetzung für Chin. Restsatz nicht erfüllt. $ggT(n, l) = 12$ damit nicht relativ prim.

c. Verallgemeinern Sie den Chinesischer Restsatz: Gesucht ist x mit $(x \mod m_i) = x_i$ und die passende Berechnungsvorschrift. Wie groß ist die Laufzeit zur Berechnung von x?

Eingabe: m_i die Module (paarweise relativ prim), x_i die gesuchten Ergebnisse mit $1 \leq i \leq n$.

Sei N_j das Produkt von $\prod_{i>0 \wedge i \neq j}^{n} m_i = m_1 \cdots m_{j-1} \cdot m_{j+1} \cdots m_n$ Sei M_i multiplikative Inverse von N_i zu m_i.

$$X = \sum_{i}^{n} x_i \underbrace{N_i M_i}_{\equiv_{m_i} 1} \tag{122}$$

$$= x_1 m_2 \cdots m_n M_1 + \ldots + x_n m_1 \cdots m_{n-1} M_n \tag{123}$$

Kosten: $T = n \cdot T_{\text{egcd}} + n(n+1)T_{\text{mult}} + nT_{\text{add}} \in \mathcal{O}(n^2)$.

6.4. RSA-Low-Exponent-Attack

a. Gg. seien die drei öffentlichen RSA-Schlüssel $(n_1 = 35, e = 3)$,$(n_1 = 35, e = 3)$,$(n_1 = 35, e = 3)$. Außerdem bekannt ist: $C_{123} = (22, 12, 216)$

Voraussetzungen:

$$C_{123} = P^3 \mod n_{123}$$

und n_{123} sind paarweise relativ prim.

Gesucht x:

$$x \mod n_1 = C_1 \wedge x \mod n_2 = C_2 \wedge x \mod n_3 = C_3 \wedge$$

$$x = \underbrace{C_1 n_2 n_3 N_{23}}_{\equiv 1 \mod n_1} + \underbrace{C_1 n_1 n_3 N_{13}}_{\equiv 1 \mod n_2} + \underbrace{C_1 n_1 n_2 N_{12}}_{\equiv 1 \mod n_3} \tag{124}$$

Suche der multiplikativen Inversen N_{123} zum Modul n_{123} mit erweiterter euklidischer Algorithmus:

$ggT(n_2 n_3, n_1) = ggT(24, 35)$:

$$35 = 1 \cdot 24 + 11 \tag{125}$$

$$24 = 2 \cdot 11 + 2 \tag{126}$$

$$11 = 5 \cdot 2 + 1 \tag{127}$$

$$1 = 11 - 5 \cdot 2 \tag{128}$$
$$1 = 35 - 24 - 5 \cdot (24 - 2 \cdot 11) \tag{129}$$
$$1 = -24 - 5 \cdot (24 - 2 \cdot (35 - 24)) \tag{130}$$
$$1 = -24 - 5 \cdot (24 + 2 \cdot 24) \tag{131}$$
$$1 = -16 \cdot 24 \tag{132}$$
$$1 = 19 \cdot 24 \tag{133}$$

$$N_{23} = 19$$

$ggT(n_1 n_3, n_2) = ggT(8, 143)$:

$$134 = 17 \cdot 8 + 7 \tag{134}$$
$$8 = 7 + 1 \tag{135}$$

$$1 = 8 - 7 \tag{136}$$
$$1 = 8 - (143 - 17 \cdot 8) \tag{137}$$
$$1 = 8 + 17 \cdot 8 \tag{138}$$
$$1 = 18 \cdot 8 \tag{139}$$

$$N_{13} = 18$$

$ggT(n_1 n_2, n_3) = ggT(160, 323)$:

$$323 = 2 \cdot 160 + 3 \tag{140}$$
$$160 = 53 \cdot 3 + 1 \tag{141}$$
$$\tag{142}$$

$$1 = 160 - 53 \cdot 3 \tag{143}$$
$$1 = 160 - 53 \cdot (323 - 2 \cdot 160) \tag{144}$$
$$1 = 107 \cdot 160 \tag{145}$$

$$N_{12} = 107$$

$$x = P^3 = 137.424.442 = 12.167(mod n_1 n_2 n_3) \tag{146}$$
$$P = \sqrt[3]{x} = \sqrt[3]{12167} = 23 \tag{147}$$

Probe:

$$23^3 \mod n_{123} = C_{123} \tag{148}$$

b.

6.5. Quadratwurzeln mod n

$$16 \mod 35$$

a. mit chin. Restsatz

Zerlegung in $pq = n$ mit $7 \cdot 5 = 35$.

Lösung von $16 \equiv 2 \mod 7$ mit Folgerung (2.5).

$$2 \frac{7+1}{4} \Rightarrow x_1 = 4 \wedge x_2 = 7 - 4 = 3$$

Lösung von $16 \equiv 1 \mod 5$:

$$x_3 = 1 \text{ und } x_4 = 4$$

$$x = 1 \mod 5 \tag{149}$$
$$x = 3 \mod 7 \tag{150}$$
$$x = 4 \mod 7 \tag{151}$$
$$x = 4 \mod 5 \tag{152}$$

Betrachtung für (149) mit (150) und (151) reicht:

$$X = x_1 pP + x_2 qQ = 1 \cdot 7P + 3 \cdot 5Q \tag{153}$$

$ggT(7,5) = 1a^{-1} = 3$ $ggT(5,7) = 1a^{-1} = 10$

$$X_1 = 4$$

$$X = x_3 pP + x_3 qQ = 4 \cdot 7P + 4 \cdot 5Q \tag{154}$$

$ggT(7,5) = 1a^{-1} = 3$ $ggT(5,7) = 1a^{-1} = 10$

$$X_2 = 9$$

Weitere

$$X_3 = 35 - X_1 = 31 \text{ und } X_4 = 35 - X_2$$

b.

[0, 1, 4, 9, 16, 25, 1, 14, 29, 11, 30, 16, 4, 29, 21, 15, 11, 9, 9, 11, 15, 21, 29, 4, 16, 30, 11, 29, 14, 1, 25, 16, 9, 4, 1]

6.6. Rabin

Sei der öffentliche Schlüssel $n = 77$, der geheime Schlüssel $p = 7$ und $q = 11$. Gegeben sei der Ciphertext $C = 23$.

a. Wie lauten die möglichen Klartexte?

Lösung für $x^2 \equiv_p C$ (2.5):

$$C^{\frac{7+1}{4}} \equiv_7 4 \tag{155}$$

$$x_{12} = 4, 3$$

Lösung für $x^2 \equiv_q C$ (2.5):

$$C^{\frac{11+1}{4}} \equiv_1 11 \tag{156}$$

$$x_{34} = 1, 10$$

Mit dem Restsatz:

$$X = (32, 45, 10, 67)$$

b. Sie wissen, dass der Klartext in seiner 7-Bit-Binärdaretsllung im höchsten Bit eine "1" hat. Welches ist der gesuchte Klartext?

$P = 67$

6.7. Elgamal

Öffentlicher Schlüssel: $p = 2579$, Primitivwurzel $g = 2, y = 2765 = 949 \mod 2579$
Geheimer Schlüssel: $x = 765$
Nachricht: $m = 1299, k = 853$

a. Führen Sie die Verschlüsselung durch.

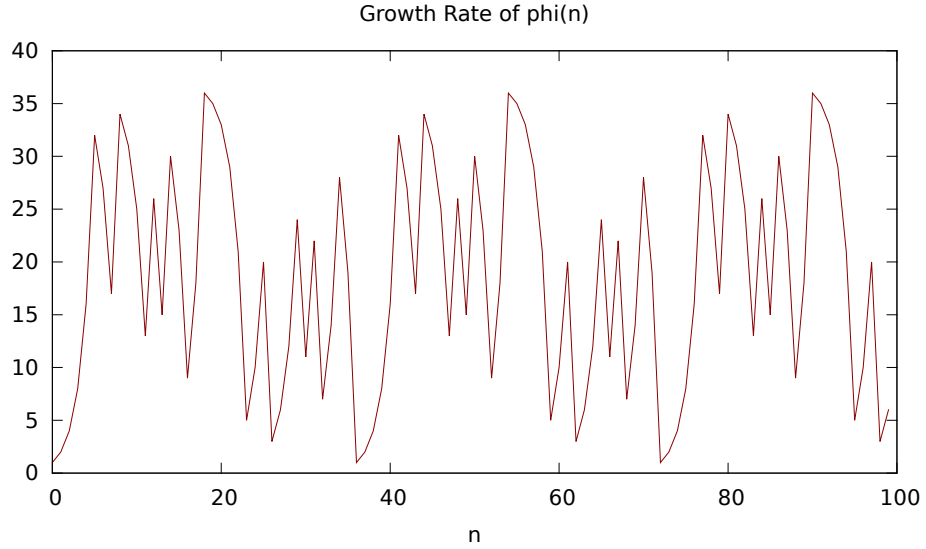
$$a = g^k \mod p = 2^{853} \mod 2579 = 435 \tag{157}$$
$$b = y^k m \mod p = 949^{853} \cdot 1299 \mod 2579 = 2396 \tag{158}$$

$$C = (435, 2396)$$

b. Führen Sie die Entschlüsselung des Ciphertexts durch und überprüfen Sie, ob Sie wieder m erhalten.

$$P = \frac{b}{a^x} \mod 2579 = \quad = \frac{2396}{436^{765}} \mod p = \frac{2396}{2424} \mod p \quad = 2396 * 2424^{-1} \mod p = 1299 \tag{159}$$

6.8. Diskrete Exponential-Funktion



6.9. Primfaktorzerlegung

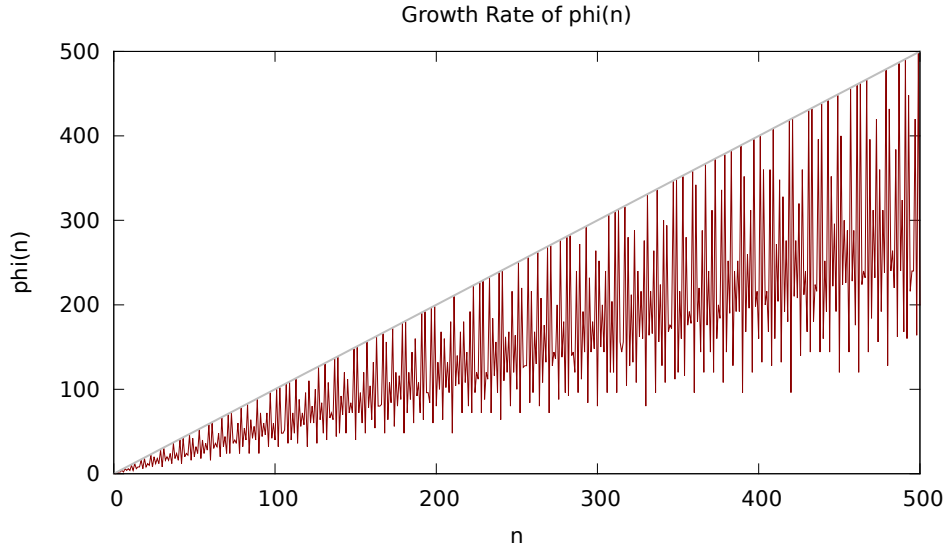
6.10. Fermatscher Primzahltest

6.11. Inverses zu $(n-1) \mod n$

6.12. $a(n-1) \mod n$

6.13. $\phi(n)$ für $n < 500$

a. Berechnen Sie $\phi(n)$ für $n < 500$ und tragen Sie die Werte in einem Graphen



b. Geben Sie eine möglichst genaue obere Schranke für $\phi(n)$ an.

$$\phi(n) \leq n - 1 \tag{160}$$
$$\in \mathcal{O}(n) \tag{161}$$

7. Digitale Signatur und Zertifikate

7.1. Keine Signatur mit dem Rucksack

7.2. RSA-Signatur

$$n = 3 * 5 = 15, e = 3, M = 7$$

$$d = 3 \Rightarrow 3 \cdot 3 \equiv 1 \mod 8$$

a. Berechnen Sie die digitale Unterschrift nach dem RSA-Verfahren.

$$D_{SK}(M) = 7^3 \mod 15 = 13 \tag{162}$$

b. Was übertragt der Sender zum Empfänger, wenn er die Nachricht M signiert übertragen will?

$$(M, D_{SK}) = (7, 13)$$

c. Verifizieren Sie die Unterschrift.

$$E_K = 13^3 \mod 15 = 7 = M$$

7.3. Elgamal-Signatur

$$p = 467 \qquad g = 2 \qquad x = 127 \qquad y = 2127 = 132 \mod 467 \tag{163}$$

Die Nachricht $M = 100$ soll unterschrieben werden. Es wird $k = 213$ verwendet.

a. Berechnen Sie die digitale Unterschrift nach dem Elgamal-Verfahren.

$$a = g^k \mod p = 2^2 13 \mod 467 = 29$$

$$M = (xa + kb) \mod (p - 1) \tag{164}$$

$$M = ((127 \cdot 29 \mod (467 - 1)) + (213 \cdot b \mod (467 - 1))) \mod (467 - 1) \tag{165}$$

$$M - 421 = 213 \cdot b \mod (467 - 1) \tag{166}$$

$$\frac{M - 421}{213} = b \mod 466 \tag{167}$$

$$(100 - 421) \cdot 213^{-1} \Rightarrow b = -321 \cdot 213^{-1} \mod 466 \tag{168}$$

$$\Rightarrow b = -321 \cdot -35 \mod 466 = 51 \tag{169}$$

b. Was übertragt der Sender zum Empfänger, wenn er die Nachricht M signiert übertragen will?

$$(M, (a, b)) = (100, (29, 51))$$

c. Verifizieren Sie die Unterschrift.

$$g^M \equiv y^a a^b \mod p \tag{170}$$
$$2^{100} \equiv 132^{29} \cdot 29^{51} \mod 467 \tag{171}$$
$$189 \equiv 189 \tag{172}$$

7.4. Länge der Passphrase für digitale Signatur

Sie haben einen 1024 Bit (2048 Bit) RSA-Schlüssel. Wie lang sollte die Passphrase zum Schutz des auf Ihrer Festplatte gespeicherten Schlüssels mindestens sein? Hinweis: Nehmen Sie an, dass sie Sicherheit eines 1024 Bit RSA-Schlüssels einem 128 Bit symmetrischen Schlüssels entspricht.

Angenommen: Passwort $w \in \Sigma^*$ besteht aus den Terminalsymbolen aus ASCII.

$$|\Sigma| = 256 - 32 = 224$$

Frage: Welche Länge muss mein Passwort haben, damit 2^{128} Möglichkeiten habe.

$$|\{w \in \Sigma^* \parallel |w| = x\}| = 2^{128}$$

$$|\Sigma|^{|w|} = 2^{128} \tag{173}$$
$$224^x = 2^{128} \tag{174}$$
$$|w| \log_2(224) = 128 \tag{175}$$
$$|w| = 16,395 \tag{176}$$

für $|\Sigma| = 64 \Rightarrow |w| = 21,3$

7.5. GPG

7.6. Signierung eines Java-Applets

7.7. PDF-Signatur

8. Kryptologische Anwendungen und Protokolle

8.1. Münzwurf am Telefon

Alice sendet $n = 34189$ an Bob. Bob wählt $x = 17209$ und sendet 4563 an Alice.

a. Alice sendet $r = 16980$ an Bob. Kann Bob n faktorisieren? Wenn ja, geben Sie die Faktorisierung an.

Ist $r = n - x$: $n - r = x = 17209$. Faktorisierung nicht möglich, Alice gewinnt

b. Alice sendet 23474 an Bob. Kann Bob n faktorisieren? Wenn ja, geben Sie die Faktorisierung an.

$ggT(x + y, n) = ggT(23474 + 17209, 34189) = 191$
o. B. d. A.: $q = \frac{n}{p} = 179, p = 191$

c. Berechnen Sie die vier Quadratwurzeln von $17209 \mod n$.

Es gibt kein x, sodass $x^2 \equiv 17209 \mod 34189$

8.2. Altersvergleich

8.3. Karten kryptologisch mischen und austeilen

Gegeben seien die folgenden vier Spielkarten und Ihre Codierung:

$$x_1 = "HerzAss" = 2 \qquad x_2 = "PikAss" = 3 \tag{177}$$
$$x_3 = "KaroAss" = 4 \qquad x_4 = "KreuzAss" = 7 \tag{178}$$

Mischen und verteilen Sie die Karten nach der in der Vorlesung beim Skat-Protokoll vorgestellten Methode an die vier Spieler A, B, C, D. Verwenden Sie folgende Permutationen in Zykelschreibweise: $\alpha = (124), \beta = (14)(23), \gamma = (134), \delta = id$ (Zykelschreibweise) Ferner sei $p = 11, a = 3, b = 7, c = 9, d = 7$.

$$K = (x_1, x_2, x_3, x_4) \tag{179}$$
$$K = (2, 4, 3, 7) \tag{180}$$
$$\tag{181}$$

Anwendung Permutation α und $x^3 \mod 11$:

$$K = (2, 3, 4, 7) \tag{182}$$
$$K = (2, 8, 9, 5) \tag{183}$$

Anwendung Permutation β und $x^7 \mod 11$:

$$K = (3, 4, 2, 7) \tag{184}$$

Anwendung Permutation γ und $x^9 \mod 11$:

$$K = (8, 3, 4, 6) \tag{185}$$

Anwendung Permutation δ und $x^7 \mod 11$:

$$K = (2, 9, 5, 8) \tag{186}$$

$$a' = 7 \qquad b' = 3 \tag{187}$$
$$c' = 9 \qquad d' = 3 \tag{188}$$

$$x_a = 2^{b'c'd'a'} \mod 11 = 7 \tag{189}$$
$$x_b = 9^{a'c'd'b'} \mod 11 = 4 \tag{190}$$
$$x_c = 5^{a'b'd'c'} \mod 11 = 3 \tag{191}$$
$$x_d = 8^{a'b'c'd'} \mod 11 = 2 \tag{192}$$
$$\tag{193}$$

8.4. Knobeln über E-Mail

Entwerfen Sie ein (dezentrales) Protokoll zum Knobeln (Schere, Stein, Papier) per E-Mail. Hinweis: Es gibt verschiedene Lösungen. Sie können z.B. Shamirs No-Key-Protokoll oder Hashfunktionen verwenden.

A		B
H(x,Stein)	$\rightarrow$	“$FSDAF”§$”§
2%!&"§$"$"§$	$\leftarrow$	H(x,Papier)
x,Stein	$\rightarrow$	verfiy
verify	$\leftarrow$	y,Papier

8.5. Chaffing and Winnowing

a. Welche Tripel (Seriennummer, Paket, MAC) übertragen Sie nach dem "Chaffing and Winnowing"-Verfahren, wenn die Nachricht "FHT" lautet und die Paketlänge einen Buchstaben lang ist?

b. Welche Tripel (Seriennummer, Paket, MAC) übertragen Sie nach dem "Chaffing and Winnowing"-Verfahren, wenn die Nachricht "F" lautet und die Paketlänge ein Bit lang ist? Verwenden Sie die 8-Bit ASCII-Codierung.

Vorgaben: Schlüssel k zur Berechnung des MAC = "geheim" Hash-Verfahren = MD5 MAC-Verfahren = $H(k, H(k, M))$ Hinweis: Sie können zur Generierung der MACS das CrypTool verwenden (Einzelverfahren =¿ Hashverfahren =¿ Generieren von MACs)

9. Kryptologische Anwendungen und Protokolle – Teil 2

9.1. Altersvergleich

Führen Sie das Protokoll zum Altersvergleich mit folgenden Parametern durch. $0 \leq a, b \leq 4$, der öffentliche RSA-Schlüssel von Bob lautet $n = 143, e = 19, x = 102$. Als Einwegfunktion verwenden Sie eine Reduktion $\mod 53$. Geben Sie die notwendigen Berechnungen, die ausgetauschten Nachrichten und das Ergebnis für folgende Alter von Alice und Bob an:

$$p = 11 \qquad q = 13 \qquad d = 19 \tag{194}$$

a. $a = 1, b = 1$

Alice		Bob
$x = 102$		
$c = E_{PK-B}(x) = 102^{19} \mod 143$		
$d = c - a = 15 - 1$	$\rightarrow$	
		$y = D_{SK-B}(14+0, 14+1, 14+2, 14+3, 14+4)$
		$= (92, 102, 42, 134, 8)$
		$z = (39, 49, 42, 28, 8)$
	$\leftarrow$	$(39, 49, 42+1, 28+1, 8+1)$
$f(x) = 49 \in z$		

b. a=1, b=3

Alice		Bob
	$\leftarrow$	$(39, 49, 42, 28, 8+1)$
$f(x) = 49 \in z$		

c. a=1, b=0

Alice		Bob
	$\leftarrow$	$(39, 49+1, 42+1, 28+1, 8+1)$
$f(x) = 49 \notin z$		

Angenommen Bob verzichtet leichtsinnigerweise auf die Anwendung der Einwegfunktion. Zeigen Sie im Fall b) wie Alice Bobs Alter rekonstruieren kann.

$$z = D_{SK-B}(14+2+1, 14+3+1, 14+4+1, 14+0, 14+1) \tag{195}$$
$$= E_{PK-B}(DSK - B(14+2+1, 14+3+1, 14+4+1, 14+0, 14+1)) \tag{196}$$
$$= (14+2+1, 14+3+1, 14+4+1, 14+0, 14+1) \tag{197}$$
$$= (2+1, 3+1, 4+1, 0, 1) \tag{198}$$
$$\text{sort} = (0, 1, 3, 4, 5) \tag{199}$$

Wert 2 fehlt, daher $b = 2$.

9.2. No-Key-Protokoll

Führen Sie das No-Key-Protokoll mit folgenden Parametern durch:

$$p = 17, a = 3, b = 5, s = 2$$

Skizzieren Sie den Protokollablauf, berechnen Sie die ausgetauschten Werte und rekonstruieren Sie das Geheimnis.

$$a' = 11 \qquad\qquad b' = 13 \tag{200}$$

$$\begin{aligned} s' &= s^{ab} \mod p && (201)\\ &= 2^{3\cdot 5} \mod 17 && (202)\\ &= 9 && (203) \end{aligned}$$

$$\begin{aligned} s &= s'^{b'a'} \mod p && (204)\\ &= 9^{13\cdot 11} \mod 17 && (205)\\ &= 2 && (206) \end{aligned}$$

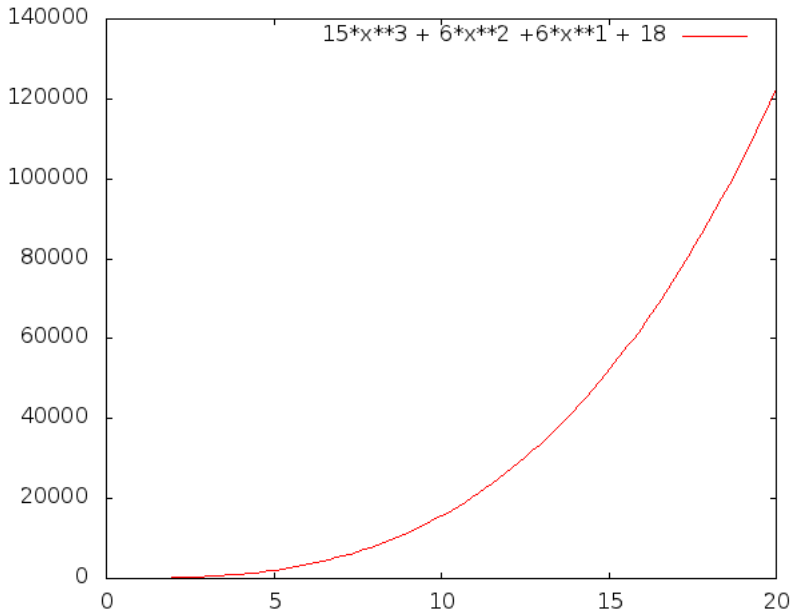
9.3. (4,6)-Schwellwertverfahren über Gleichungssystem

Es seien $p = 19$ und die folgenden Wertepaare gegeben: $(1, 7), (2, 3), (3, 4), (16, 4), (17, 5)$ und $(18, 1)$. Rekonstruieren Sie aus $(1, 7), (2, 3), (17, 5)$ und $(18, 1)$ das Geheimnis und das Polynom durch Lösen des entsprechenden linearen Gleichungssystems. Zeichnen Sie das Polynom für die Wert von $x = 0$ bis $x = 20$.

4 Freiheitsgrade = Polynomd 3. Grades: $a_3x^3 + a_2x^2 + a_1x^1 + a_0 = y$. Gesucht Koeffizienten a_i:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 8 & 4 & 2 & 1 \\ 4913 & 289 & 17 & 1 \\ 5832 & 324 & 18 & 1 \end{pmatrix} \cdot \begin{pmatrix} a_3 \\ a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} 7 \\ 3 \\ 5 \\ 1 \end{pmatrix} \tag{207}$$

$$a = \begin{pmatrix} 15 \\ 6 \\ 6 \\ 18 \end{pmatrix} \Rightarrow 15x^3 + 6x^2 + 6x^1 + 18 \tag{208}$$



9.4. (3,4)-Schwellwertverfahren über Lagrange

Gegeben sind folgende Punkte (1,6) (2,3), (3,2), (4,3). Wählen Sie drei Punkte aus und rekonstruieren Sie das Polynom über das Lagrangesche-Interpolationspolynom. Was ist das Geheimnis? Zeichen Sie das Polynom.

10. Übungsaufgaben: Kryptologische Anwendungen und Protokolle – Teil 3

10.1. Fiat-Shamir

Gegeben sei $n = 35, s = 29$.Die Anzahl der Runden sei $k = 5$. Peggy wählt für die Runden 1 bis 5 folgende Zufallszahlen: $r_1 = 9, r_2 = 13, r_3 = 7, r_4 = 12, r_15 = 17$ Victor antwortet mit folgender Bit-Folge: $b_{12345} = 0, 1, 0, 0, 1$

a. Führen Sie Fiat-Shamir durch. Geben Sie $x_1, y_1, \ldots, x_5, y_5$ an.

Peggy	Kanal	Viktor
$x = r_1^2 \mod n =$	$11 \rightarrow$	
	$\leftarrow 0$	b_1
$y = r_1 \cdot s^b \mod n$	$5 \rightarrow$	$y^2 = x \cdot v^b \mod n = 9^2 = 11$ stimmt
$x = r_2^2 \mod n =$	$29 \rightarrow$	
	$\leftarrow 1$	b_2
$y = r_2 \cdot 29^1 \mod n$	$1 \rightarrow$	$y^2 = x \cdot v^b \mod n \Rightarrow 1 = 29^2 = x$ stimmt
$x = r_3^2 \mod n =$	$14 \rightarrow$	
	$\leftarrow 0$	b_3
$y = r_3 \cdot s^0 \mod n$	$7 \rightarrow$	$y^2 = x \cdot v^b \mod n \Rightarrow 7^2 = 14$ stimmt
$x = r_4^2 \mod n =$	$4 \rightarrow$	
	$\leftarrow 0$	b_4
$y = r_4 \cdot s^0 \mod n$	$12 \rightarrow$	$y^2 = x \cdot v^b \mod n \Rightarrow 12^2 = 4$ stimmt
$x = r_5^2 \mod n =$	$9 \rightarrow$	
	$\leftarrow 1$	b_5
$y = r_5 \cdot 29^1 \mod n$	$3 \rightarrow$	$y^2 = x \cdot v^b \mod n \Rightarrow 3^2 = 9 \cdot 1$ stimmt

b. Angenommen eine Angreiferin Eve errät die Bitfolge. Wie hoch ist die Wahrscheinlichkeit dafür?

Gegen Sie unterschiedliche Pärchen für x, y an, mit der sich Eve gegenüber Victor als Peggy ausgeben kann.

$$|\mathcal{B}|^5 = 32 \text{ Möglichkeiten} \tag{209}$$

$$\begin{matrix} x & y \\ r_1 & r_1^2 \end{matrix} \tag{210}$$

10.2. Bit-Commitment mit Einweg-Hash-Funktion

Sei die Hashfunktion $H = MD5, R_1 = 01000\ 11010$ und $R_2 = 00110\ 11110$.

a. Führen Sie das Bit-Commitment-Protokoll (Festlegung und Offenlegung) für $b = 0$ und $b = 1$ durch.

Eingabe folgte als ASCII:

```
echo -n 010001101000110111100 | md5sum
    50b3ce3f3b19bf4ec7f358a977d9c7e4
echo -n 010001101000110111101 | md5sum
    cda9c8f61cd6d2b35d778af367b9a467
```

b. Angenommen Alice verrät R_2. Wie kommt Bob allein durch den Festlegungsteil des Protokolls an das von Alice gewählte Bit?

Er berechnet jeweils $H(R_1, R_2, 0, 1)$ und sieht anhand des Hashes welches Bit Alice gewählt hat.

10.3. Elektronisches Geld

a. Protokoll 4: Ein Betrüger möchte eine Bank dazu bringen, blind eine 100€-Münze zu signieren, seinem Konto aber nur 1€ zu belasten. Dazu erzeugt er 99 Münzen à 1€ und eine à 100€. Wie groß ist die Wahrscheinlichkeit, dass die Bank blind die 100€-Münze signiert?

Urnenmodell ohne zurücklegen:

Wie groß ist die Wahrscheinlichkeit eine rote Kugel (100€-Münze) aus 100 Kugeln bei 99 Zügen auszuwählen.

$$P(X = m) = \frac{\binom{M}{m}\binom{N-M}{n-m}}{\binom{N}{n}} \tag{211}$$

$$N = 100 \qquad n = 99 \qquad M = 1 \qquad m = 1 \tag{212}$$

$$P(X = 1) = \frac{\binom{1}{1}\binom{99}{99}}{\binom{100}{99}} \tag{213}$$

$$= \frac{1 \cdot 1}{100} = 0.01 \tag{214}$$

b. Wie viele Bits muss die Seriennummer einer E-Münze lang sein, damit die Wahrscheinlichkeit für eine zufällige Übereinstimmung zweier Münzen kleiner 10^{10} ist?

Geburtstagsparadoxon:

$$10^{10} = 10^9 * 10 = 10^{3^3} * 10 \approx 2^1 0^3 * 2^3 = 2^3 3$$

Antwort mind. 66 Bits.

c. Protokoll 5: Wenn Eve elektronische Münzen von Alice stiehlt, kann sie damit noch nicht bezahlen. Warum?

Aufgrund der Identitätsabfragen des Händlers.

d. Protokoll 5:An welcher Stelle im Protokoll hat Eve trotzdem leichtes Spiel, wenn sie es schafft, münzen zu stehlen?

Gute Frage nächste Frage....

e. Protokoll 5: Alice kopiert eine Münze, verwendet diese zwei Mal und der Händler steht als Betrüger da. Durch welche Festsetzung kann die Wahrscheinlichkeit dafür kleiner 10^{10} gehalten werden?