

Summary_MOVK

January 21, 2020

Contents

1 SW01 - Mathematical Basis	4
1.1 3 Computing the (multiplicative) inverse I¶	4
1.1.1 Solution	4
1.2 5 Computing the (multiplicative) inverse III (Fermats Theorem)	4
1.2.1 Solution	4
2 SW02 - Mathematical Basis II	4
2.1 1: Generator or primitive element of a group	4
2.2 2. Field	5
2.2.1 Solution	5
2.3 3. The Galois Field $GF(2^2)$	5
2.4 5. Quadratic congruence (Legendre)	6
2.4.1 Solution	6
3 SW03 - Secret-key or symmetric cryptography	8
3.1 1. DES S-box S_3	8
3.1.1 Solution	8
3.2 4. AES S-box	8
3.2.1 Solution	8
4 SW04 - Cryptographic Utilities	9
4.1 4 LFSR	9
4.1.1 Solution	9
4.1.2 Tabelle zum herausfinden des Clocks	13
4.2 5: Probability for a collision (find p)	14
4.2.1 Solution	14
4.3 6: Probability for a collision (find n)	14
4.3.1 Solution	14
5 SW05 - Public Key Cryptography I	14
5.1 1 Shamir's three-pass protocol	14
5.1.1 Solution	15
5.2 2 Diffie Hellman	16
5.2.1 Solution	16
5.3 3 Discrete Logarithm Problem	17
5.3.1 Solution	17

5.4	4.0 - RSA Ablauf	18
5.5	4.1 Attack on textbook RSA (factorizing n)	19
5.5.1	Solution	19
5.6	5 Attack on textbook RSA — small exponent e	19
5.6.1	Solution	20
5.7	6 Attack on textbook RSA — common module n	21
5.7.1	Solution	21
5.8	8 Elgamal 2nd	22
5.8.1	Solution	22
6	SW06 - Public Key Cryptography II	23
6.1	1 Elliptic curve over real numbers \mathbb{R}	23
6.1.1	Solution	23
6.2	2 Elliptic curve over real numbers \mathbb{R}	24
6.2.1	Solution	24
6.3	4 Elliptic curve over finite field \mathbb{Z}_7	25
6.3.1	Solution	25
6.4	5 Diffie-Hellman using ECC	26
6.4.1	Solution	27
6.4.2	Kurz Erklärung (reicht für MEP)	27
6.5	6 Elliptic Curve Discrete Logarithm Problem (ECDLP)	27
6.5.1	Solution	27
7	SW07 - Protokolle I (Ladan)	28
7.1	1: Diffie-Hellmann-Schlüsselaustausch	28
7.1.1	Solution	28
7.2	3: Blinde Signatur	29
7.2.1	Solution	29
7.3	Aufgabe 4: Bit-Commitment	30
7.3.1	Solution	30
7.4	4.1 Bit-Commitment (Beispiel)	31
7.4.1	Solution	31
8	SW08 - Protokolle II	32
8.1	Aufgabe 1: Dining-Cryptographers	32
8.1.1	Solution	32
8.2	Aufgabe 3: MIXe	33
8.2.1	Solution	33
8.3	Aufgabe 5: Secret Splitting / Geteiltes Geheimnis	34
8.3.1	Solution	34
8.4	Aufgabe 6: (2,3)-Schwellenwertproblem - (Optional)	34
8.4.1	Solution	34
8.5	Aufgabe 8: Threshold-Verfahren (Optional 2)	34
8.5.1	Solution	34
9	SW09 - Zertifikate und Public-Key-Systeme	35
9.1	Aufgabe 1: Zertifikate	35
9.1.1	Solution	35

9.2 Aufgabe 2: Zertifikatshierarchie	35
9.2.1 Solution Quick	35
10 SW10 - Homomorphe Verschlüsselung	36
10.1 Aufgabe 1: Homomorphe Verschlüsselung	36
10.1.1 Solution	36
10.2 Aufgabe 2: Homomorphie-Eigenschaft von EL-GAMAL	36
10.2.1 Solution	37
10.3 Aufgabe 3: Paillier-Verfahren	39
10.3.1 Solution	39
11 SW11 - eVoting	41
11.1 Aufgabe 2: Wahlprotokoll mit symmetrischer Verschlüsselung	41
11.1.1 Solution	41
11.2 Aufgabe 3: additives homomorphes Wahlprotokoll	41
11.2.1 Solution	42
12 SW12 - ePayment	42
12.1 Aufgabe 2: Random Serialnumber	42
12.1.1 Solution	43
12.2 Aufgabe 4: Serialnumber Bits	43
12.2.1 Solution	43
12.3 Aufgabe 5: Secret Splitting	44
12.3.1 Solution	44
13 SW13 - Blockchain	44
13.1 Aufgabe 1: Blockchain	44
13.1.1 Solution	44
13.2 Aufgabe 2: Elektronische Zahlungsverkehr	44
13.2.1 Solution	44
13.3 Aufgabe 3: Bitcoin	45
13.3.1 Solution	45
14 SW14 - Quanten Kryptografie	46
14.1 Aufgabe 2: BB84	46
14.1.1 Solution	46
14.2 Aufgabe 3: Again BB84	47
14.3 Aufgabe 4: How to find the period of an injective function f	47
14.3.1 Solution	47
14.4 Aufgabe 5: Shor's algorithm to factor 35	48
14.4.1 Solution	48

1 SW01 - Mathematical Basis

1.1 3 Computing the (multiplicative) inverse I¶

Your Task: Find the multiplicative inverses of a modulo n (if it exists) if
1. $a = 5, n = 13$ 2. $a = 7, n = 15$ 3. $a = 5, n = 15$

1.1.1 Solution

a = 5, n = 13

$$5^{-1} = 5 \cdot x = 1 \pmod{13} \Rightarrow x = 8$$

Denn: $(8 \cdot 5) \pmod{13} \equiv 40 \pmod{13} = 1 \pmod{5} = 1$

a = 7, n = 15

$$7^{-1} = 7 \cdot x = 1 \pmod{15} \Rightarrow x = 13$$

Denn: $13 \cdot 7 \pmod{15} = 91 \pmod{15} = 1$

a = 5, n = 15

$$5^{-1} = 5 \cdot x = 1 \pmod{15} \Rightarrow x = \text{NULL}$$

Denn: 5^{-1} hat kein modulares inverses zu 15, da sie nicht **teilefremd** sind.

1.2 5 Computing the (multiplicative) inverse III (Fermats Theorem)

Your Task: Compute the multiplicative inverse of 9 modulo 11.

1.2.1 Solution

It states, that for any prime p which is not a divisor of a , the following holds

$$a^{p-1} \equiv 1 \pmod{p} \text{ i.e. } a^{p-1} \equiv_p 1$$

Fermats Theorem (mod inverse)

Das Theorem besagt ausserdem, falls $a^{p-1} \equiv 1 \pmod{p}$ gilt, dann ist $a^{p-2} \equiv a^{-1} \pmod{p}$ - also gleich das modular Inverse.

Also: $9^{11-2} \pmod{11} \equiv 5$

2 SW02 - Mathematical Basis II

2.1 1: Generator or primitive element of a group

Is 3 a generator of $(\mathbb{Z}_{11}^*, \cdot)$?

Find a generator of $(\mathbb{Z}_{11}^*, \cdot)$

Wir erinnern uns: ein Generator enthält alle Elemente einer (modularen) Gruppe indem wir Ihn immer wieder mit sich selbst potenzieren.

$3^x \pmod{11} : (3, 9, 5, 4, 3, 9, 5, \dots)$ - (3 wiederholt sich nach 4 Zahlen und ist somit kein Generator)

$2^x \pmod{11} : (2, 4, 8, 5, 10, 9, 7, 3, 6, 1)$ - (2 ist ein Generator von \mathbb{Z}_{11}^*)

2.2 2. Field

Show that if p is prime, then \mathbb{Z}_p together with addition and multiplication modulo p constitutes a field. Check whether the rules (i)-(ix) hold?

2.2.1 Solution

Gemäss dem Field Summary erfüllt \mathbb{Z}_p all diese bedingungen:

		\mathbb{N}	\mathbb{Z}	\mathbb{Q}	\mathbb{R}	\mathbb{C}	\mathbb{Z}_p
(i)	$(a + b) + c = a + (b + c)$	✓	✓	✓	✓	✓	✓
(ii)	$a + b = b + a$	✓	✓	✓	✓	✓	✓
(iii)	$a + 0 = a$	✓	✓	✓	✓	✓	✓
(iv)	$a + (-a) = 0$	X	✓	✓	✓	✓	✓
(v)	$(a \cdot b) \cdot c = a \cdot (b \cdot c)$	✓	✓	✓	✓	✓	✓
(vi)	$a \cdot b = b \cdot a$	✓	✓	✓	✓	✓	✓
(vii)	$a \cdot 1 = a$	✓	✓	✓	✓	✓	✓
(viii)	$a \cdot a^{-1} = 1$	X	X	✓	✓	✓	✓
(ix)	$a \cdot (b + c) = a \cdot b + a \cdot c$	✓	✓	✓	✓	✓	✓

Wir testen das ganze mit "p = 7"

Wichtig ist vor allem die Regel viii. Bei einer Primzahl p gibt es für jedes a ein Inverses, ansonsten nicht.

2.3 3. The Galois Field $GF(2^2)$

Complete the following tables for addition (top) and multiplication (bottom).

+

+	00	01	10	11
00	00	01	10	11
01	01	10	11	00
10	10	11	00	01
11	11	00	01	00

*

.	00	01	10	11
00	00	00	00	00
01	00	01	10	11
10	00	10	11	01
11	00	11	01	10

Note: Etwas wie Sudoku

2.4 5. Quadratic congruence (Legendre)

Does the linear congruence $x^2 \equiv 446 \pmod{1129}$ have a solution x ?

You need not compute x ; just decide, if a solution exists. Use the Legendre symbol

$$\left(\frac{446}{1129}\right)$$

2.4.1 Solution

Quadratic Congruence (Legendre) ①

$x^2 \equiv 446 \pmod{1129}$

Legendre Schreibweise: $\left(\frac{446}{1129}\right)$

① $\left(\frac{446}{1129}\right) = 446^{\frac{p-1}{2}} = 446^{\frac{1129-1}{2}} = 446^{\text{sqd}} \pmod{1129}$ (Umformung)

② Berechnen mit SQM oder Divide & Conquer

$SG4 = 1000110100$
 $= \cancel{Q} \cancel{M} \cancel{Q} \cancel{Q} \cancel{Q} \cancel{M} \cancel{Q} \cancel{M} \cancel{Q} \cancel{Q}$

↓

$$\begin{aligned}
 & \left(\left(\left(\left(\left(\left(446^2 \right)^2 \right)^2 \right)^2 \right)^2 \right)^2 \cdot 446 \right) \cdot \left(\left(\left(\left(\left(\left(446^2 \right)^2 \right)^2 \right)^2 \right)^2 \right)^2 \cdot 446 \right) \cdot \left(\left(\left(\left(\left(\left(446^2 \right)^2 \right)^2 \right)^2 \right)^2 \right)^2 \right)^2 \mod 1129 \\
 & \quad \text{(2)} \\
 & \quad \downarrow \\
 & \quad \frac{212}{915^2} \\
 & \quad \downarrow \\
 & \quad \frac{367^2}{338 \cdot 446} \\
 & \quad \downarrow \\
 & \quad \frac{591^2}{420 \cdot 446} \\
 & \quad \downarrow \\
 & \quad \frac{1035^2}{933^2} \\
 & \quad \downarrow \\
 & \quad \frac{30 \cdot 446}{961^2} \\
 & \quad \downarrow \\
 & \quad 1128^2 \Rightarrow 1
 \end{aligned}$$

3 SW03 - Secret-key or symmetric cryptography

3.1 1. DES S-box S_3

The input to the DES S-box S_3 is 110111. What's the output?

3.1.1 Solution

The first and last bit (here 11) referring to the row index 3 and the middle four bits (here 1011) referring to the column index 11.

The output is 0011.

S_3		Mittlere 4 Bits des Eingabewertes																
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Äußere Bits	0	00	1010	0000	1001	1110	0110	0011	1111	0101	0001	1101	1100	0111	1011	0100	0010	1000
	1	01	1101	0111	0000	1001	0011	0100	0110	1010	0010	1000	0101	1110	1100	1011	1111	0001
	2	10	1101	0110	0100	1001	1000	1111	0011	0000	1011	0001	0010	1100	0101	1010	1110	0111
	3	11	0001	1010	1101	0000	0110	1001	1000	0111	0100	1111	1110	0011	1011	0101	0010	1100

3.2 4. AES S-box

If we input the byte 11011101 into the AES S-box, what's the output?

3.2.1 Solution

We split 11011101 in half, this gives us 1101 / 1101 → row / column

1101 = 13 and therefore the output is C1

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f	
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76	
10	ca	b2	c9	7d	fa	59	47	f0	ad	d4	a2	af	9e	a4	72	c0	
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15	
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75	
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84	
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf	
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8	
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2	
80	cd	0c	13	ee	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73	
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db	
10	a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
11	b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
12	c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
13	d0	70	3e	b5	66	48	03	f6	0	61	35	57	b9	86	c1	1d	9e
14	e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
15	f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Spaltenindex (erste 4 Bit): 11011101 == Spalte 13

Reihenindex (nächste 4 Bit): 11011101 == Reihe 13

4 SW04 - Cryptographic Utilities

4.1 4 LFSR

Gegeben: - LFSR Länge: 5 - Output: 1011001010

4.1.1 Solution

LSFR (2)

Gegeben: • LFSR Länge 5
• Output: 10110 01010

Gesucht: • Periode von LSFR
• Polynom
• Maximale Periode, LSFR 5

① Grundform LSFR Polynom der Länge 5:

$$C(D) = 1 + c_1 \cdot D + c_2 \cdot D^2 + c_3 \cdot D^3 + c_4 \cdot D^4 + c_5 \cdot D^5$$

② Initial Bits: Die ersten 5 Octet-Bits in umgekehrter Reihenfolge: 01101

③ And-Gateways festlegen

Wir füllen die Initial Bits in das Polynom ein. Der Output entspricht dann den nächsten 5 Bits der Octet-Bits

$$c_1 \cdot 0 + c_2 \cdot 1 + c_3 \cdot 1 + c_4 \cdot 0 + c_5 \cdot 1 = 0$$

Das Octet Bit 0 schieben wir jetzt von links in unsere Initial-Bits für die nächste Sequenz. Das rechteste Bit fliegt raus. →

③ Neue BitReihenfolge: 00110 X

Neue Sequenz ins Polynom:

$$c_1 \cdot 0 + c_2 \cdot 0 + c_3 \cdot 1 + c_4 \cdot 1 + c_5 \cdot 0 = 1$$

$$\begin{array}{r} \overset{12345}{(01010)} \\ \underline{-} \\ \begin{array}{r} \leftarrow \end{array} \end{array}$$

Das machen wir jetzt 5 mal (solange wir Output Bits haben) :

$$c_1 \cdot 0 + c_2 \cdot 1 + c_3 \cdot 1 + c_4 \cdot 0 + c_5 \cdot 1 = 0$$

$$\begin{array}{r} \swarrow \\ c_1 \cdot 0 + c_2 \cdot 0 + c_3 \cdot 1 + c_4 \cdot 1 + c_5 \cdot 0 = 1 \end{array}$$

$$\begin{array}{r} \swarrow \\ c_1 \cdot 1 + c_2 \cdot 0 + c_3 \cdot 0 + c_4 \cdot 1 + c_5 \cdot 1 = 0 \end{array}$$

$$c_1 \cdot 0 + c_2 \cdot 1 + c_3 \cdot 0 + c_4 \cdot 0 + c_5 \cdot 1 = 1$$

$$c_1 \cdot 1 + c_2 \cdot 0 + c_3 \cdot 1 + c_4 \cdot 0 + c_5 \cdot 0 = 0$$

Mit dieser Gleichung kann man jetzt 'erraten', welches die Gateways sind $\rightarrow c_1, c_3, c_5$

Je nach LSF R ist das aber fast unmöglich. Ander MFP wird ein einfacheres LSF R kommen \rightarrow siehe nächste Seite

32 LFSR Länge 4: 00011110

$$c_1 \cdot 1 + c_2 \cdot 0 + c_3 \cdot 0 + c_4 \cdot 0 = 1$$

$$\cancel{c_1 \cdot 1} + c_2 \cdot 1 + c_3 \cdot 0 + c_4 \cdot 0 = 1$$

$$c_1 \cdot 1 + c_2 \cdot 1 + c_3 \cdot 1 + c_4 \cdot 0 = 1$$

$$c_1 \cdot 1 + c_2 \cdot 1 + c_3 \cdot 1 + c_4 \cdot 1 = 0$$

Hier sehen wir sehr schnell das c_1 ein Gateway ist.

c_2 und c_3 können keine Gateways sein und c_4 ist wieder eins.

4 Mit den Gateways könnte man jetzt die Periode des LFSR suchen.

Dafür müsste man die Tabelle erstellen und schauen, wann sich die Input Bits wiederholen (Dauert lange ...)

$$1+1+0=0 \bmod 2$$

Clock	S ₄	S ₃	S ₂	S ₁	S ₀	Out
0	0	1	1	0	1	-
1	0	0	1	1	0	1
2	1	0	0	1	1	0
	0	1	0	0	1	1

Nach wievielen Clocks wiederholt sich "01101" = Periode

⑤ Maximale Periode: Die maximale Periode

ist nur mit "primitive Polynoms" möglich \Rightarrow Siehe Tabelle!

Für LSFN-S wäre das $(0, 2, 5)$ $S = \text{lange LSFN}$

Also wenn c_2 und c_5 Gateways sind

Die Formel lautet dann: $2^L - 1 \Rightarrow 2^5 - 1 = 31$

- ## ⑥ Mit den Gateways c_1, c_3, c_5 können wir das effektive Polynom aufschreiben.

Aus: $C(D) = 1 + c_1 \cdot D + c_2 \cdot D^2 + c_3 \cdot D^3 + c_4 \cdot D^4 + c_5 \cdot D^5$

Wird: $C(D) = 1 + D^1 + D^3 + D^5$

4.1.2 Tabelle zum herausfinden des Clocks

Gegeben:

- LFSR Länge = S
- Output: $1011001010\dots$

Vorgehen:

- Tabelle erstellen
- Output einfügen
- Shift nach oben links und S₄ alles abföhren
- S₄ ist das Ergebniss des AND Gateways der oberen Zeile.
So kann das Gateway herausgefunden werden.
z.B. $0 \oplus 1 \oplus 1 = 0$
Gateway = 10101
- Restliche Tabelle ausfüllen

Clock	S ₄	S ₃	S ₂	S ₁	S ₀	Out
0	0	1	1	0	1	(0)
1	0	0	1	1	0	1
2	1	0	0	1	1	0
3	0	1	0	0	1	1
4	1	0	1	0	0	1
5	0	1	0	1	0	0
6	0	0	1	0	1	0
7	1	0	0	1	0	1
8	1	1	0	0	1	0
9	0	1	1	0	0	1
10	1	0	1	1	0	0
	1	1	0	1	1	0

4.2 5: Probability for a collision (find p)

There are 40 people in a room. You bet, that there are at least 2 people with the same birthday. What is the probability, that You win? Use the exact formula as well as the approximation.

4.2.1 Solution

Allgemein Formel für W'Keit (p) von Kollision berechnen.

$$p = 1 - e^{\frac{-n(n-1)}{(2*m)}} = 1 - e^{\frac{-40(40-1)}{(2*365)}} = 0.882$$

n = Anzahl effektiver Werte (z.B. Personen)

m = Anzahl möglicher Werte (z.B. Tage im Jahr)

p = Wahrscheinlichkeit

4.3 6: Probability for a collision (find n)

Suppose You use a hash function of length 128 bits. How many hash values would You have to compute in order to find a collision with probability at least 90%?

$$n = 2^{(m+1)/2} \cdot \sqrt{(\ln(\frac{1}{1-p}))}$$

where m=128 and p=0.9.

Angenäherte Formel: $n = 2^{\frac{m}{2}}$

4.3.1 Solution

$$n = 2^{(m+1)/2} \cdot \sqrt{(\ln(\frac{1}{1-p}))} = 2^{(128+1)/2} \cdot \sqrt{(\ln(\frac{1}{1-0.9}))} = 3.958 \cdot 10^{19}$$

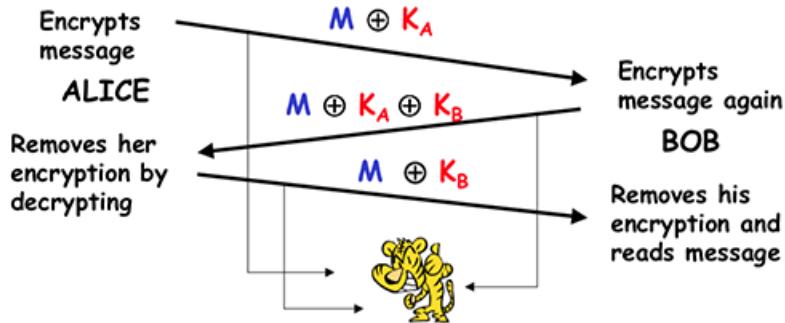
5 SW05 - Public Key Cryptography I

5.1 1 Shamir's three-pass protocol

Alice and Bob want the implement Shamir's three-pass protocol using the Vernam cipher, i.e. one-time pad. This is supposed to provide perfect secrecy. Is the following protocol secure?

Vernam ciphers (one-time pads) commute!

$$(M \oplus BSS\#1) \oplus BSS\#2 = (M \oplus BSS\#2) \oplus BSS\#1$$



Your Task: Can You compute the message? Make an example with $M = 010110111101$, $K_A = 101101110100$, and $K_B = 001011011011$.

5.1.1 Solution

Gegeben: - Shamir Protocol

- $M: 010110111101$
- $K_A: 101101110100$
- $K_B: 001011011011$

1. Encrypt $M \oplus K_A$

M	0	1	0	1	1	0	1	1	1	1	0	1
K_A	1	0	1	1	0	1	1	1	0	1	0	0
C_A	1	1	1	0	1	1	0	0	1	0	0	1

XOR it \oplus

2. Encrypt again - $M \oplus K_A \oplus K_B$

C_A	1	1	1	0	1	1	0	0	1	0	0	1
K_B	0	0	1	0	1	1	0	1	1	0	1	1
C_B	1	1	0	0	0	0	0	1	0	0	1	0

XOR

3. Remove K_A (Encrypt again $C_B \oplus K_A$)

C_B	1	1	0	0	0	0	0	1	0	0	1	0
K_A	1	0	1	1	0	1	1	1	0	1	0	0
C_A	0	1	1	1	0	1	1	0	0	1	1	0

XOR

4. Remove K_B , Get M (Encrypt $C_B \oplus K_B$)

C_B	0	1	1	1	0	1	1	0	0	1	1	0
K_B	0	0	1	0	1	1	0	1	1	0	1	1
M	0	1	0	1	1	0	1	1	1	1	0	1

✓

5.2 2 Diffie Hellman

Alice and Bob agree to use $n = 13$ and $e = 11$. Alice chooses her secret number $a = 5$, whereas Bob chooses $b = 7$.

Your Task: What are the requirements for n and e ? Are they fulfilled? Describe the key agreement protocol step by step using the above assumptions about a and b . What is the common secret key?

5.2.1 Solution

e must be a generator for \mathbb{Z}_n^* , this is true for $e = 11$ to $n = 13$

Diffie Hellmann

Gegeben: $n = 13$ / modulo (auch p)

$e = 11$ / generator in \mathbb{Z} (auch g)

Alice $a = 5$ / Secret Keys
Bob $b = 7$

① n muss eine Primzahl sein

$$e \Rightarrow 1 < e < n-1$$

→ e muss Generator in \mathbb{Z}_n sein

②

$$A = e^a \mod n = 11^5 \mod 13 = 7$$

⇒ Alice sendet A an Bob

$$B = e^b \mod n = 11^7 \mod 13 = 2$$

⇒ Bob sendet B an Alice

Alice berechnet gemeinsamen Schlüssel

$$K_{AB} = B^a \mod n = e^{ba} \mod n = 2^5 \mod 13 = 6$$

Bob berechnet gemeinsamen Schlüssel

$$K_{AB} = A^b \mod n = e^{ab} \mod n = 7^7 \mod 13 = 6$$

5.3 3 Discrete Logarithm Problem

Assume Mallory intercepts the message $A = 9$ from Alice to Bob and $B = 3$ from Bob to Alice. He also knows $n = 13$ and $g = 11$.

Your Task: Suppose Mallory wants to know the common key. Describe his steps to find this key!

5.3.1 Solution

He would need to find a such that $A = g^a \pmod{n}$. Then he could calculate $B^a = K \rightarrow (g^{b \cdot a})$.

$$9 = 11^a \pmod{13}$$

$$11^1 = 11$$

...

$$11^8 = 9 - > a = 8 \rightarrow K = A^b = 3^8 \pmod{13} = 9$$

For large numbers, this is **infeasable**.

5.4 4.0 - RSA Ablauf

RSA 1-Ablauf

- ① Wähle zwei Primzahlen p und q | $p=7, q=11$
- ② Berechne $n = p \cdot q$ | $n = 7 \cdot 11 = 77$
- ③ Berechne $\phi n = (p-1)(q-1)$ | $\phi n = (7-1)(11-1) = 60$
- ④ Wähle eine zu ϕn teilerfremde Zahl $e \Rightarrow \gcd(e, \phi n) = 1$
 $e = 13$
- ⑤ Berechne Mod-Inverse von e mod ϕn Mit EEA
 $d \cdot e = 1 \text{ mod } \phi n$ | $d \cdot 13 = 1 \text{ mod } 60$
 $d = 37$
- ⑥ Private Key = d
- ⑦ Public Key = (n, e)

5.5 4.1 Attack on textbook RSA (factorizing n)

The public key $(n, e) = (2537, 13)$ was used to encrypt the plaintext M . Eve intercepts the ciphertext $C = 2081$.

Your Task: Show how Eve computes the plaintext M !

5.5.1 Solution

RSA 2-Factorizing n Cifer Text

Gegeben: $n = 2537$ $C = 2081$ $e = 13$

An Mer \Rightarrow kleines n gegeben (2537)

- ① n Faktorisieren
 $n = 2537 = 43 \cdot 59 = p \cdot q$
- ② $\phi(n)$ berechnen
 $\phi(n) = (p-1) \cdot (q-1) = (43-1) \cdot (59-1) = 2436$
- ③ Mod-Inverse d berechnen
 $d \cdot e = 1 \pmod{\phi(n)} \Rightarrow d \cdot 13 = 1 \pmod{2436}$
 $d = 937$ // mit Euklid berechnet
- ④ Decrypt Cifer C
 $M = C^d \pmod{n} = 2081^{937} \pmod{2537} = 1819$

5.6 5 Attack on textbook RSA — small exponent e

Frequently, the exponent e in the public key (n, e) is chosen very small, say $e = 3$. Hence, encryption of m is very fast

$$c = m^3 \pmod{n}$$

Your Task: Assume the message m is sent to 3 different people using textbook RSA, with moduli $n_1 = 377$, $n_2 = 391$, and $n_3 = 589$. You get hold of the corresponding ciphertexts

$$330 = m^3 \pmod{377}$$

$$34 = m^3 \pmod{391}$$

$$419 = m^3 \pmod{589}$$

Compute $m = \sqrt[3]{x}$ using the CRT, where $x = m^3$ satisfies the system of linear congruences

$$x \equiv 330 \pmod{377}$$

$$x \equiv 34 \pmod{391}$$

$$x \equiv 419 \pmod{589}$$

5.6.1 Solution

$$n_1 = 377, n_2 = 391, \text{ and } n_3 = 589$$

$$x_1 = 330 \pmod{377}, x_2 = 34 \pmod{391}, x_3 = 419 \pmod{589}$$

Solution Steps Gegeben: $n_1, n_2, n_3 \mid x_1, x_2, x_3$

Schritt 1

N_1, N_2, N_3 berechnen.

$$N_1 = n_2 \cdot n_3$$

$$N_2 = n_1 \cdot n_3$$

$$N_3 = n_1 \cdot n_2$$

Schritt 2

y_1, y_2, y_3 berechnen.

$$y_1 = \text{mod-Inverse } N_1 \text{ mod } x_1$$

$$y_2 = \text{mod-Inverse } N_2 \text{ mod } x_2$$

$$y_3 = \text{mod-Inverse } N_3 \text{ mod } x_3$$

Schritt 3

x berechnen

$$x = \text{Summe}(x_1 \cdot N_1 \cdot y_1 + x_2 \cdot N_2 \cdot y_2 + x_3 \cdot N_3 \cdot y_3 \pmod{(n_1 \cdot n_2 \cdot n_3)})$$

Schritt 4

e-te Wurzel von x berechnen = m

$$\sqrt[3]{x} = m$$

5.7 6 Attack on textbook RSA — common module n

Suppose the CTO of a company wants that all employees use the same module n . The individual employees have pairwise different (e_i, d_i) . Suppose, two employees A and B have the public keys (n, e_A) and (n, e_B) where $\gcd(e_A, e_B) = 1$.

Now the administration sends the encrypted message m to the two employees

$$c_A = m^{e_A} \pmod{n}$$

$$c_B = m^{e_B} \pmod{n}$$

Your Task: Design an example with small numbers which demonstrates, this attack!
Assume $n = 11 \cdot 13$, i.e. $p = 11$ and $q = 13$.

5.7.1 Solution

RSA 4 - Gemeinsames n

ATTACK on RSA A & B verwenden dasselbe n

Gegeben: $n = 143$ $e_A = 13$
 $m = 10$ $e_B = 7$

$c_A = m^{e_A} = 10^{13} \pmod{143} = 10$

$c_B = m^{e_B} = 10^7 \pmod{143} = 10$

Calculate a, b so that: $a \cdot e_A + b \cdot e_B = 1$

$a \cdot 13 + b \cdot 7 = 1$
 $\hookrightarrow a = -1, b = 2$

$c_A^a \cdot c_B^b = 10^{-1} \cdot 10^2 = 10^1 = 10 = \underline{\underline{m}}$

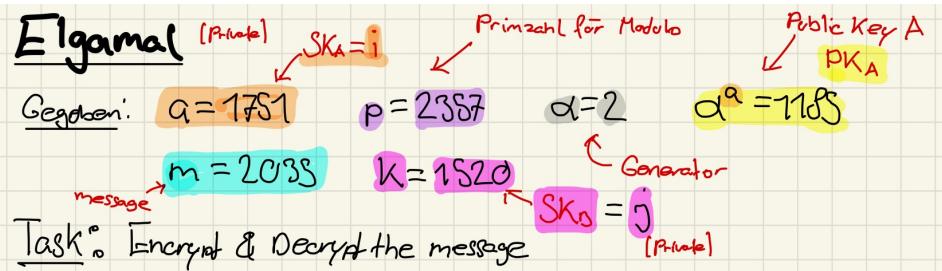
~~Rekord~~ Weil: $c_A^a \cdot c_B^b = m^{e_A \cdot a + e_B \cdot b} = m^1 = m$

5.8 8 Elgamal 2nd

Alice uses the private key $a = 1751$ and computes the public key ($p = 2357, \alpha = 2, \alpha^a = 1185$). Now Bob wants to encrypt the message $m = 2035$. He uses the random $k = 1520$.

Your Task: Compute the encrypted message and show how Alice decrypts the message.

5.8.1 Solution



ALICE

Sendet $\boxed{\text{PK}_A, \alpha, p}$ an Bob

Bob

$$\text{SK}_B = j \in [\alpha : p-2] = j \in [2 : 2357-2] \quad \text{Secret Key B}$$

$$\text{SK}_B = 1520$$

$$\text{PK}_B = \beta = \alpha^j = 2^{1520} \mod p = 2^{1520} \mod 2357 = 1430$$

$$\text{Session Key: } K_m = \text{PK}_A^j \mod p = 1185^{1520} \mod 2357 = 2084$$

$$\text{Cipher Text: } y = m \cdot K_m \mod p = 2035 \cdot 2084 \mod 2357 = 697$$

Sendet $\boxed{j, \text{PK}_B}$ an Alice

ALICE

$$\text{Session Key: } K_m = \text{PK}_B^{SK_A} = \beta^i = 1430^{1751} \mod 2357 = 2084$$

$$\text{Message: } m = K_m^{-1} \cdot y \mod p = 872 \cdot 697 \mod 2357 = \underline{\underline{2035}}$$

$872 \downarrow$ ← erweiterter Euklid

Der letzte Schritt K_m^{-1} mit erweitertem Euklid berechnen ($a \cdot 2084 + b \cdot 2357 = 1$)

6 SW06 - Public Key Cryptography II

6.1 1 Elliptic curve over real numbers \mathbb{R}

Given the elliptic curve $E : y^2 = x^3 - 3x + 9$. If this curve is not singular (check this), compute a and b , such that $P_1 = (0, a)$ and $P_2 = (2, b)$ lay on E . Compute $P = P_1 + P_2$ and draw E together with P_1 , P_2 and P .

6.1.1 Solution

1st Step: Always Check if its Not-Singular:

$$4a^3 + 27b^2 \neq 0 \text{ (not singular)}$$

$$4 * (-3)^3 + 27 * 9^2 = -4 * 27 + 27 * 81 = 27 * (81 - 4) \neq 0$$

Elliptic Curve over \mathbb{R}

- Gegeben:
- $E : y^2 = x^3 - 3x + 9$
 - $P_1(0, a) \parallel P_2(2, b)$

$$P_1 \neq P_2$$

Gesucht: $a, b, R = P_1 + P_2$

M

$$y^2 = 0^3 - (3 \cdot 0) + 9 = 9 \quad | \quad y_1 = \sqrt{9} = 3$$

$$\underline{P_1 = (0, 3)}$$

+ Jeder Punkt kann an x-Achse gespiegelt werden
 $-P_1 = (0, -3)$



P2

$$y^2 = 2^3 - (3 \cdot 2) + 9 = 11 \quad | \quad y_2 = \sqrt{11} = 3,316$$

$$\underline{P_2 = (2, 3,316)}$$

weil
 $P_1 \neq P_2$

$R = P_1 + P_2$

$$S = \frac{y_2 - y_1}{x_2 - x_1} = \frac{3,316 - 3}{2 - 0} = 0,158$$

$$x_3 = S^2 - x_1 - x_2 = 0,158^2 - 0 - 2 = 0,025 - 2 = -1,975$$

$$y_3 = S \cdot (x_1 - x_3) - y_1 = 0,158 \cdot (0 - (-1,975)) - 3 \\ = 0,158 \cdot (1,975) - 3$$

$$\underline{R = (-1,975, -2,688)}$$

6.2 2 Elliptic curve over real numbers \mathbb{R}

Given the elliptic curve $E : y^2 = x^3 - 3x + 5$ and point $P = (2, 2.65) \in E$.

Your Task: Draw E , point P and compute $2P, 4P$ and $8P$. Solve this problem with minimal computational work!

6.2.1 Solution

Elliptic Curve over \mathbb{R} II

Gegeben: $E: y^2 = x^3 - 3x + 5$

$P(2, 2.65)$

Achtung!
Grenzen
Fazit

Task: - Compute $2P, 4P, 8P$

$2P$

$$S_2 = \frac{3x_1^2 + a}{2y_1} = \frac{3 \cdot 2^2 + (-3)}{2 \cdot 2.65} = \frac{9}{5.3} = 1.698$$

x_{2P} = $S_2^2 - x_1 - x_2 = 1.698^2 - 2 - 2 = -1.17$

y_{2P} = $S_2(x_1 - x_{2P}) - y_1 = 1.698 \cdot (2 - (-1.17)) - 2.65 = 2.73$

$2P(-1.17, 2.73)$

$4P$ (aus $2P$)

$$S_4 = \frac{3x_{2P}^2 + a}{2y_{2P}} = \frac{3 \cdot (-1.17)^2 + (-3)}{2 \cdot 2.73} = 0.202$$

x_{4P} = $S_4^2 - x_{2P} - x_{4P} = 0.202^2 - (-1.17) - (-1.17) = 2.38$

y_{4P} = $S_4(x_{2P} - x_{4P}) - y_{2P} = 0.202 \cdot (-1.17 - 2.38) - 2.73 = -3.44$

$4P(2.38, -3.44)$

... Analog für $8P$ mit $4P$

6.3 4 Elliptic curve over finite field \mathbb{Z}_7

Let $E : y^2 \equiv x^3 + 3x + 2 \pmod{7}$ an elliptic curve over \mathbb{Z}_7 .

Your Task: 1. Compute all points on E over \mathbb{Z}_7 . 2. What is the order of the group? (Hint: Do not miss the identity element O) 3. Given the element $\alpha = (0, 3)$, determine the order of α . Is α a primitive element?

6.3.1 Solution

Elliptic Curve over Finite Field \mathbb{Z}_7

①

$E : y^2 \equiv x^3 + 3x + 2 \pmod{7}$

$x = 1$ bis 7 in die Gleichung einsetzen und rechnen
+ $y = 7$ rechnen

$x = 1$	$x = 0$
$y^2 = 6$, $y = 1$ (no sqrt)	$y^2 = 2$
	$\hookrightarrow 2 + 7 = 9 \Rightarrow \sqrt{9} = 3$
	$\hookrightarrow 9 + 7 = 16 \Rightarrow \sqrt{16} = 4$
	$\Rightarrow (0, 3), (0, 4)$

$x = 2$

$y^2 = 16$, $y = 4 (\sqrt{16})$, $y = 3 (16 - 7 = 9 \Rightarrow \sqrt{9} = 3)$

$x = 3$

$y^2 = 38$, $y = 1$ (no sqrt for 38, 31, 24, 17, 10, 3)

$x = 4$

$y^2 = 78$, $y = 78, 71, 69, \dots, 36 \Rightarrow y = 1, 6$

$x = 5$

$y^2 = 142$, $y = 121, \dots, 100 \Rightarrow y = 4, 3$

$x = 6$	$P : (0, 3); (0, 4); (2, 3); (2, 4); (4, 1); (4, 6); (5, 3); (5, 4)$
$y = 1$	Die Ordnung ist 8 (Punkte) + 1 (Nullpunkt) = 9

Elliptic Curve over Finite Field \mathbb{Z}_7

(2)

$$E: y^2 \equiv x^3 + \frac{3}{a}x + \frac{2}{b} \pmod{7}$$

$$\alpha = P(0, 3)$$

α is a primitive Element if all Points in \mathbb{Z} can be calculated with it (yes)

Alle Punkte berechnen $(2\alpha, 3\alpha, 4\alpha \dots)$

$2P$

$$\begin{aligned} S &= (3x_1^2 + a) \cdot (2y_1)^{-1} \pmod{p} \quad // \text{ weil } p_1 = p_2 \\ &= (3 \cdot 0^2 + 3) \cdot 6^{-1} \pmod{7} = 3 \cdot 6 = 18 \pmod{7} = \underline{\underline{4}} \end{aligned}$$

$$x_3 = S^2 - x_1 - x_2 = 4^2 - 2 \cdot 0 = 16 \pmod{7} = \underline{\underline{2}}$$

$$y_3 = S \cdot (x_1 - x_2) - y_1 = 4 \cdot (0 - 2) - 3 = -11 \pmod{7} = \underline{\underline{3}}$$

$$\alpha 2 = P_2 = \underline{\underline{(2, 3)}}$$

$$P_1(0, 3) \quad P_2(2, 3)$$

$$\alpha 3 = P_3 = P_1 + P_2$$

$$S = (y_2 - y_1) \cdot (x_2 - x_1)^{-1} \pmod{p} \quad // \text{ da } P_1 \neq P_2 !$$

$$= (3 - 0) \cdot (2 - 0)^{-1} = 3 \cdot 2^{-1} = 3 \cdot 4 = \underline{\underline{0}}$$

$$x_3 = S^2 - x_1 - x_2 = 0^2 - 0 - 2 = \underline{\underline{-2}} \pmod{7} = \underline{\underline{5}}$$

$$y_3 = S \cdot (x_1 - x_2) - y_1 = 0 \cdot (0 - 2) - 3 = -3 \pmod{7} = \underline{\underline{4}}$$

$$P_3 = \underline{\underline{(5, 4)}}$$

... etc für alle Punkte ...

6.4 5 Diffie-Hellman using ECC

Alice and Bob agree to use the elliptic curve $E : y^2 = x^3 + x + 1$ and point $P = (5, 4) \in E$. Alice chooses her secret number $a = 5$, whereas Bob chooses $b = 7$.

Your Task: Describe the key agreement protocol step by step using the above assumptions about a and b . What is the common secret key?

Suppose that the elliptic curve is in mod 23 ($23 = n$) and all points $\in \mathbb{Z}$

6.4.1 Solution

$$1 \cdot P = (5, 4)$$

$$2 \cdot P = (17, 20)$$

$$4 \cdot P = (13, 7)$$

$$5 \cdot P = (17, 3) = Q_a$$

$7 \cdot P$ is 0 because whether it's calculated $2P + 5P$, or $P + 6P$, or $3P + 4P$, x is always equal in both points.

Erklärung Da die x-Werte von $(2P \& 5P)$, $(P \& 6P)$, $(3P \& 4P)$ gleich sind, würde bei der Berechnung $0^{-1} \text{mod } p$ rauskommen \rightarrow Von 0 gibt es aber kein Modulares Inverses, weshalb $7P = 0$.

Alice computes $Q_{a \cdot b} = a \cdot Q_b$	Bob computes $Q_{a \cdot b} = b \cdot Q_a$
$Q_{a \cdot b} = 5 \cdot Q_b$	$Q_{a \cdot b} = 7 \cdot Q_a$
$Q_{a \cdot b} = 4 \cdot Q_b + Q_b$	$Q_{a \cdot b} = 4 \cdot Q_a + Q_a + 2 \cdot Q_a$
Alice gets $Q_{a \cdot b} = 0$	Bob gets $Q_{a \cdot b} = 0$

6.4.2 Kurz Erklärung (reicht für MEP)

1. Zuerst müssten mit $1P$ alle anderen Punkte berechnet werden ($2P, 3P, 4P \dots 7P$) {Dauert viel zu lange...}
2. Alice berechnet A und sendet es an Bob: $A = a \cdot P = 5 \cdot P = 5P = (17, 3)$
3. Bob berechnet B und sendet es an Alice: $B = b \cdot P = 7 \cdot P = \mathbb{O}$
4. Alice kann den Common-Key berechnen: $K_{BA} = a \cdot B = 5 \cdot B = 5 \cdot 7P = 35P = 5 \cdot \mathbb{O} = \mathbb{O}$
5. Bob kann auch den Common-Key berechnen: $K_{AB} = b \cdot A = 7 \cdot A = 7 \cdot 5P = 35P = \mathbb{O}$

6.5 6 Elliptic Curve Discrete Logarithm Problem (ECDLP)

Assume Mallory intercepts the message $A = (10, 6)$ from Alice to Bob and $B = (7, 11)$ from Bob to Alice. He also knows the elliptic curve $E : y^2 \equiv x^3 + 2x + 2 \pmod{17}$, which forms a cyclic group of order 19, and point $P = (5, 1)$.

Your Task: Suppose Mallory wants to know the common key. Describe his steps to find this key!

6.5.1 Solution

n	1	2	3	4	5	6	7	8	9	10
nG	(5, 1)	(6, 3)	(10, 6)	(3, 1)	(9, 16)	(16, 13)	(0, 6)	(13, 7)	(7, 6)	(7, 11)

n	11	12	13	14	15	16	17	18	19
nG	(13, 10)	(0, 11)	(16, 4)	(9, 1)	(3, 16)	(10, 11)	(6, 14)	(5, 16)	\emptyset

Kurzbeschreibung (für MEP)

1. Mit der Kurvenformel und $P1$ alle anderen Punkte bis $P19$ berechnen. (dauert wieder ewigs)
2. In der Tabelle schauen, welche P 's den Punkten *Alice* : $(10, 6)$ und *Bob* : $(7, 11)$ entsprechen $\rightarrow P3$ und $P10$

3. Voila: Secret Key Alice=3, Bob=10

7 SW07 - Protokolle I (Ladan)

7.1 1: Diffie-Hellmann-Schlüsselaustausch

Im Unterricht wurde das Protokoll Diffie-Hellmann sicher vorgeführt. Kann ein Angreifer Namens Mr. X das System angreifen, falls er die Zahlen α und β kennen würde? Begründen Sie Ihre Antwort.

7.1.1 Solution

α und β sind A und B , sprich die öffentlichen Teile des Schlüssels.

Das Knacken des Diffie-Hellmann-Schlüsselaustauschprotokolls ist gleichwertig zum Berechnen des diskreten Logarithmus und somit nicht in vernünftiger Zeit lösbar.

7.2 3: Blinde Signatur

Führen Sie zu Zweit die blinde Signatur durch. Protokollieren Sie das Vorgehen und zeigen Sie, wie der Signierer die Nachricht berechnen kann, welche er (blind) signiert!

7.2.1 Solution

Blinde Signatur

Key length 7bit

Alice

Allg. Bekannt: $n = 7 \cdot 11 = 77$

$\lambda_n = \lambda(77) = \text{kGV}(\varphi(7), \varphi(11))$
 $= \text{kGV}(6, 10) = \underline{30}$

e : Primzahl / publ. key
 $e = 17$

d : $d \equiv e^{-1} \pmod{\lambda_n}$ / priv. key
 $= 17^{-1} \pmod{30} = \underline{23}$

$t = M \cdot K^e \pmod{n} =$
 $18 \cdot 9^{17} \pmod{77} = \underline{72}$

>>> sendet t an Bob

Bob

Signiert t

$U = t^d \pmod{n} = 72^{23} \pmod{77} = \underline{18}$

>>> sendet U an Alice

Alice

entschlüsselt Signatur

$M^d = U \cdot K^{-1} \pmod{n} = 18 \cdot 9^{-1} \pmod{n} = 18 \cdot 60 \pmod{77} = \underline{2}$

$M^d = 18^{23} \pmod{77} = \underline{2}$

INFO: Ich habe hier mit kGV berechnet, das ist nicht falsch (die Rechnung geht auf) aber für das wahre RSA und für unsere Prüfung müssen wir $\varphi(n)$ -> Also $(p-1) \cdot (q-1)$ verwenden. Hier die $\varphi(n)$ Variante:

Blinde Signatur v2

Alice

$$k = 9 \quad / \text{Random (teilerfremd zu } n)$$

$$M = 18 \quad / \text{Message}$$

$$t = M \cdot k^e \bmod n = \\ 18 \cdot 9^{13} \bmod 77 = 43$$

Allg. Bekannt:

RSA-Teil



Prime

$$n = 7 \cdot 11 = 77$$

$$\varphi n = (7-1) \cdot (11-1) = 60$$

$$e: 1 < e < \varphi n$$

$$\gcd(\varphi n, e) = 1$$

$$e = 13$$

$$d: d = e^{-1} \bmod \varphi n \quad / \text{privkey} \\ = 13^{-1} \bmod 60 = 37$$

>>> sendet t an Bob

Bob

Signiert t

$$u = t^d \bmod n = 43^{37} \bmod 77 = 43$$

>>> sendet u an Alice

Alice

entschlüsselt Signatur

$$M^d = u \cdot k^{-1} = 43 \cdot 9^{-1} = 43 \cdot 60 \bmod 77 = 39$$

\uparrow
mod inverse

$$M^d = 18^{37} \bmod 77 = 39 \quad (\text{Double Check})$$

7.3 Aufgabe 4: Bit-Commitment

Alice leitet die Vertriebsabteilung einer IT-Firma. Sie bereitet eine Offerte mit ihrem Team vor, um an einem digitalen Ausschreibungsprozess teilzunehmen. Als Protokoll zum Einreichen der Offerten wurde Bit-Commitment, allerdings mit einem Bit b der Länge eins (1), festgelegt. Wie kann Alice sicher sein, dass Bob (ein Mitbewerber) das Bit b nicht berechnen kann?

7.3.1 Solution

Bei der Festlegungsphase des Bits b verwendet das Bit-Commitment-Protokolls eine kollisionsfreie Einweg-Hash-Funktion. Dadurch kann Alice sicher sein, dass der Mitbewerber Bob nicht das Bit b (das Urbild unter der kollisionsfreien Einweg-Hash-Funktion) aus dem Funktionswert berechnen kann.

7.4 4.1 Bit-Commitment (Beispiel)

7.4.1 Solution

Bit Commit

1.) Wähle primes p und q | $p=5, q=7$

$$1.1.) n = p \cdot q = 5 \cdot 7 = 35$$

2.) Wähle m mit Jacobi/Legendre Eigenschaft "1" (Quadratic Non residue)

$$\left(\frac{m}{n}\right) \Rightarrow \left(\frac{36}{35}\right) = 36^{\frac{p-1}{2}} = 36^{\frac{35-1}{2}} = 36^{17} \bmod 35 = 1 \quad \checkmark$$

3.) Wähle Bit b | $b=1$

4.) Wähle Zufallszahl r | $r=6$

5.) Berechne $c = f(b, r) = m^b \cdot r^2 \bmod n$

$$= 36^1 \cdot 6^2 \bmod 35 = 1$$

\Rightarrow Alice sendet c an Bob

6.) Offenlegung: Alice sendet b und r an Bob

Bob rechnet nach ob c korrekt ist

$$c = 1 = 36^1 \cdot 6^2 \bmod 35 = 1$$

8 SW08 - Protokolle II

8.1 Aufgabe 1: Dining-Cryptographers

Ziehen Sie das Protokoll Dining-Cryptographers im Falle einer Dreiergruppe einige Male durch, indem sie das Protokoll mit einem Schiedsrichter überprüfen:

8.1.1 Solution

David, Pascal, Stefan:

Geheimes Bit erzeugen

$$b_{DP} = 1, b_{DS} = 1$$

$$b_{PD} = 1, b_{PS} = 0$$

$$b_{SD} = 1, b_{SP} = 0$$

XOR der jeweiligen Geheimen Bits berechnen

$$\text{David: } b_{DP} + b_{DS} \bmod 2 = 1 + 1 \bmod 2 = 0$$

$$\text{Pascal: } b_{PD} + b_{PS} \bmod 2 = 1 + 0 \bmod 2 = 1$$

$$\text{Stefan: } b_{SP} + b_{SD} \bmod 2 = 0 + 1 \bmod 2 = 1$$

Wenn der Kryptologe nicht bezahlen will, wird das XOR-Resultat bekannt gegeben, möchte er bezahlen wird das Gegenteil vom XOR-Resultat ausgegeben

David hat "0". David bezahlt nicht und meldet deshalb "0"

Pascal hat "1", Pascal bezahlt und meldet deshalb das Gegenteil, also "0"

Stefan hat "1". Stefan bezahlt nicht und meldet deshalb "1"

Alle Ergebnisse nochmals XОРen

David + Pascal + Stefan mod 2 = 0 + 0 + 1 = 1 (Das bit ist 1, also hat 1 (gütiger) Kryptologe bezahlt.

8.2 Aufgabe 3: MIXe

Beispiel MIXe

8.2.1 Solution

MIXe

Gesucht: Versand einer Nachricht über 3 MIXe

Anonym

Definitionen: MIXe T_1 mit Pub/Priv Keys (m, k_1)

$T_2 (p_2, k_2)$

$T_3 (p_3, k_3)$

Nachricht m

Sender x | Empfänger z

Nachrichtenfluss: $x \rightarrow T_1 \rightarrow T_2 \rightarrow T_3 \rightarrow z$

① Sender definiert: $x_1 = E_{p_1}(E_{p_2}(E_{p_3}(m|z)|T_3)|T_2)$

↳ sendet an T_1

② T_1 entschlüsselt: $D_{k_1}(E_{p_1}(E_{p_2}(E_{p_3}(m|z)|T_3)|T_2))$

$= E_{p_2}(E_{p_3}(m|z)|T_3)|T_2$

③ T_1 sendet weiter an T_2

T_2 entschlüsselt und sendet weiter an T_3

T_3 entschlüsselt und sendet weiter an z (m)

8.3 Aufgabe 5: Secret Splitting / Geteiltes Geheimnis

Verteilen Sie das Geheimnis 01011001 auf drei Leute! Protokollieren Sie das Vorgehen und zeigen Sie, wie man das Geheimnis rekonstruiert.

8.3.1 Solution

$$M = 01011001$$

$$\text{Alice: } 11001100 = A$$

$$\text{Bob: } 10101010 = B$$

$$\text{Cyril: } A \oplus B \oplus M = 00111111 = C$$

$$M = A \oplus B \oplus C = 01011001$$

8.4 Aufgabe 6: (2,3)-Schwellenwertproblem - (Optional)

Das Geheimnis M wird mit den drei Zufallszahlen R_1, R_2, R_3 in drei Teilgeheimnisse aufgeteilt.

Das Teilgeheimnis $M_1 = (M_{11}, M_{12}, M_{13}) = (R_1 \oplus M, R_2, R_3)$ ist gegeben.

1. Definieren Sie M_2 und M_3 .
2. Rekonstruieren Sie \mathbf{M} nur mit zwei Teilgeheimnissen

8.4.1 Solution

Ist ähnlich wie die Aufgabe 5.

Wir definieren z.B. $M_2 = (M_{21}, M_{22}, M_{23}) = (R_2 \oplus M, R_3, R_1)$

Im Endeffekt haben wir etwas wie $M_{11} = R_1 \oplus M$ und ebenfalls $M_{23} = R_1 \oplus M$

Wir machen $M_{11} \oplus M_{23}$, so löst sich R_1 auf und wir haben sofort wieder M .

8.5 Aufgabe 8: Threshold-Verfahren (Optional 2)

Alice teilt ein Geheimnis S mit einem Polynom des 2. Grades mit. Die Zahl S ist die y-Achsenabschnitt $(0, S)$, die das Geheimnis darstellt. Als Teilgeheimnisse sind drei verschiedene Punkte $(3, 2), (4, 1), (5, 2)$ bekannt. Wie rekonstruieren Sie das Geheimnis?

8.5.1 Solution

Allgemeines Polynom zweiten Grades:

$$f(x) = a_2 \cdot x^2 + a_1 \cdot x^1 + a_0 \cdot 1$$

$$2 = a_2 \cdot 3^2 + a_1 \cdot 3^1 + a_0 \cdot 1$$

$$1 = a_2 \cdot 4^2 + a_1 \cdot 4^1 + a_0 \cdot 1$$

$$2 = a_2 \cdot 5^2 + a_1 \cdot 5^1 + a_0 \cdot 1$$

$$\begin{bmatrix} 2 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 9 & 3 & 1 \\ 16 & 4 & 1 \\ 25 & 5 & 1 \end{bmatrix} \cdot \begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix}$$

Dann hier etwas Linalg Python Magic, was an der MEP nicht möglich ist :)

$$\begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 1 \\ -8 \\ 17 \end{bmatrix}$$

Und da $S = a_0$ haben wir damit die Lösung $S = 17$.

9 SW09 - Zertifikate und Public-Key-Systeme

9.1 Aufgabe 1: Zertifikate

Erstellen Sie ein kurzes Protokoll zur Zertifikaterstellung und Zertifikatüberprüfung.

9.1.1 Solution

1. Hierzu erzeugt der Kunde im ersten Schritt auf seiner eigenen privaten Hardware ein Schlüsselpaar (einen privaten Schlüssel und einen öffentlichen).
2. Der Kunde erzeugt eine CSR-Datei, welche ein elektronisches Formular ist. Es enthält neben den Antragsdaten auch seinen öffentlichen Schlüssel.
3. Im dritten Schritt sendet der Kunde die CSR an die CA.
4. Die CA prüft den Antrag (also die CSR-Datei mit den Formularangaben und dem enthaltenden öffentlichen Zertifikat). Bei positiver Prüfung sendet die CA dem Käufer ein neues öffentliches Zertifikat zurück (als doppelt signierter öffentlicher Schlüssel).

Für die Zertifikatsprüfung kann nun der öffentliche Schlüssel der von Alice bereitgestellt wird von einer Drittperson mit dem öffentlichen Schlüssel der signing CA verglichen.

9.2 Aufgabe 2: Zertifikatshierarchie

1. Welche Vorteile hat eine Zertifikatshierarchie für eine Firma und für die einzelne Benutzerin?
2. Welche Nachteile hat eine Zertifikatshierarchie für eine Firma und für den einzelnen Benutzer?

9.2.1 Solution Quick

1. Zertifikatsbasierte Authentifizierungen vereinfachen das gesamte Authentifizierungsverfahren. Durch eine eigene CA hat man in der Firma volle Kontrolle über die Zertifikate und muss nicht über eine public CA gehen.
2. Das Problem sind vor allem die Trusts mit anderen Firmen/CA's. Hierfür muss immer jeweils das Root Zertifikat der anderen Firmen im Truststore eingespielt werden.

10 SW10 - Homomorphe Verschlüsselung

10.1 Aufgabe 1: Homomorphe Verschlüsselung

1. Welches der drei Verschlüsselungsverfahren (RSA, EL-GAMAL, Paillier) ist eher geeignet für die Verwendung in homomorpher Verschlüsselung eingesetzt werden? Begründen bitte Ihre Antwort?
2. Geben Sie ein Beispiel, in welchem Bereich das Paillier-Verfahren eingesetzt werden könnte? Begründen Sie Ihre Antwort?

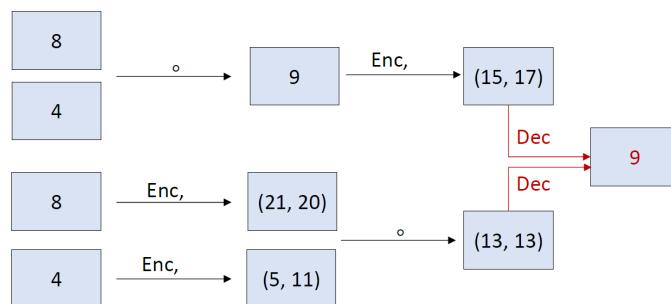
10.1.1 Solution

Name	Art des Verfahrens	Art der Homomorphie	Klartext Operation	Geheimtext Operation
RSA	deterministisch	mehrheitlich	.	.
EL-GAMAL	probabilistisch	mehrheitlich	.	.
Paillier	probabilistisch	additiv	+	.

1. RSA ist semantisch nicht korrekt, da durch Polluting (rauschen) das Ver+Entschlüsseln nicht immer dasselbe ergeben. Konkret wenn das Rechenergebnis grösser als der Modulus ist, wird das Resultat "abgeschnitten". Elgamal und Paillier sind besser geeignet da sie semantisch sind. Jedoch ist Elgamal nur für die Multiplikation und Paillier nur für die Addition geeignet (kommt halt auf den Einsatzzweck an). Paillier ist ausserdem sehr komplex und rechenintensiv.
2. Paillier könnte z.B. für E-Voting genutzt werden. So könnte man alle abgegebenen Stimmen zählen (addieren) ohne zu sehen wer konkret wen gewählt hat

10.2 Aufgabe 2: Homomorphie-Eigenschaft von EL-GAMAL

Berechnen Sie bitte jeden Schritt im unteren Teil des Beispiels (s. unten). Die Berechnung für den oberen Teil finden Sie auf den Folien SW10



10.2.1 Solution

Homomorph ElGamal

①

1. Schlüsseldefinition

$$n = 23 \quad (\text{Prime}) \quad \} \quad \text{public Key Teil 1,2}$$

$$g = 7 \quad (\text{Generator}) \quad \}$$

$$x = 5 \quad (\text{Zufallszahl}, 1 < x < n-1) \quad \} \quad \text{private Key}$$

$$y = g^x \bmod n = 7^5 \bmod 23 = 17 \quad \} \quad \text{public Key Teil 3}$$

$$\text{Public Key } \text{pk} = (23, 7, 17)$$

2. Verschlüsselung

$$m_1 = 8, m_2 = 4 \quad | \quad r_1 = 3, r_2 = 7 \quad \} \quad \text{Zufallszahl}$$

$$\text{Enc}(m) = (g^r, m \cdot y^r) \bmod n$$

$$\text{Enc}(8) = (7^3, 8 \cdot 17^3) \bmod 23 = (21, 20) \quad / \text{mit } r_1$$

$$\text{Enc}(4) = (7^7, 4 \cdot 17^7) \bmod 23 = (5, 11) \quad / \text{mit } r_2$$

3. Entschlüsseln

$$\text{Decr}(c_1, c_2) = c_2 \cdot (c_1^{-1}) \bmod n$$

$$\text{Decr}(21, 20) = 20 \cdot (21^{-1} \bmod 23) = 20 \cdot 5 \bmod 23 = 8$$

$$\text{Decr}(5, 11) = 11 \cdot (5^{-1} \bmod 23) = 11 \cdot 15 \bmod 23 = 4$$

4. Homomorphe Enc + Decr

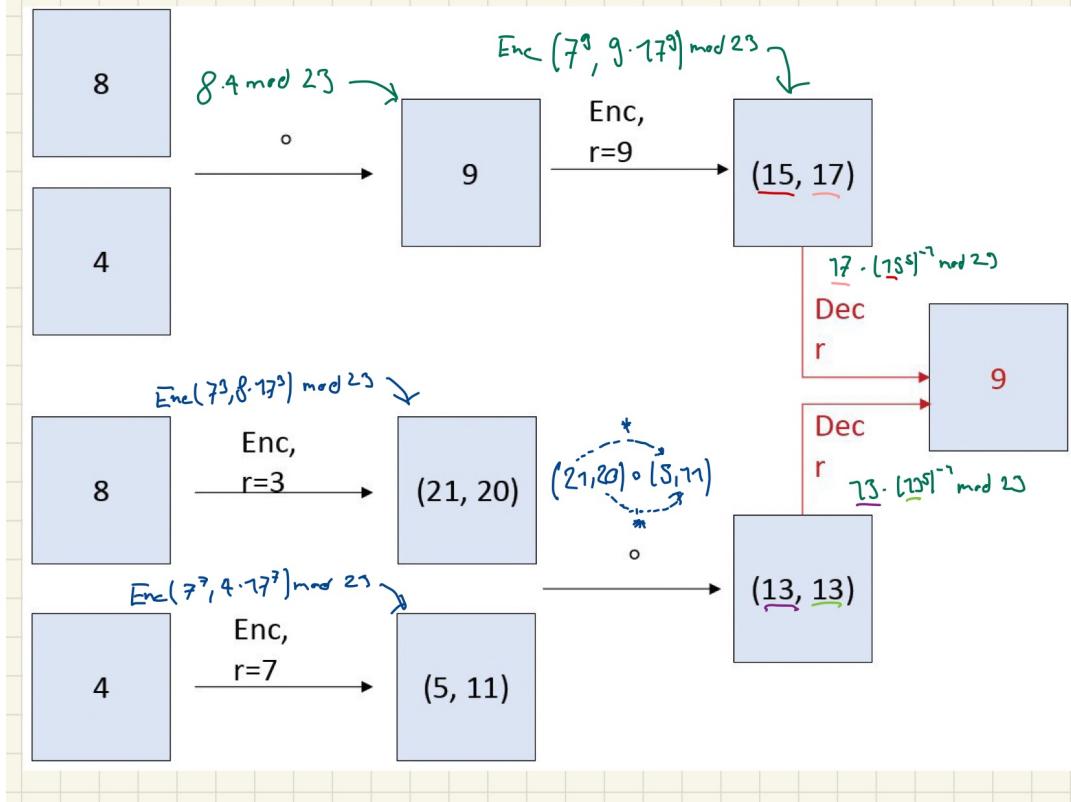
(2)

$$c(8) \circ c(4) = (21, 20) \circ (5, 11) \bmod 23$$

$$= ((21 \cdot 5 \bmod 23), (20 \cdot 11 \bmod 23)) = (13, 13)$$

$$\text{Decr}(13, 13) = 13 \cdot (13)^{-1} \bmod 23 = 13 \cdot 6 \bmod 23 = 9$$

Das entspricht der Klartextrechnung : $m_1 \cdot m_2 \bmod n$
 $= 8 \cdot 4 \bmod 23 = 9$



10.3 Aufgabe 3: Paillier-Verfahren

Gegeben sind zwei Primzahlen $p=3$ und $q=5$. Sei $g=16$ aus \mathbb{Z}_{225}^* zufällig gewählt. 1. Berechnen Sie jeweils den öffentlichen und privaten Schlüssel. 2. Verschlüsseln Sie den Klartext $m=13$. 3. Entschlüsseln Sie den Ciphertext $c=71$.

10.3.1 Solution

Paillier

Gegeben: $p=3$, $q=5$, g aus \mathbb{Z}_{225}^* ($g=16$)

1. Schlüsselerzeugung

$$n = p \cdot q = 3 \cdot 5 = 15$$

$$n^2 = 15^2 = 225$$

$$\lambda = \text{KgV}(p-1, q-1) = \text{KgV}(2, 4) = 4$$

$$\text{PublicKey } pk = (n, g) = (15, 16)$$

$$\text{PrivateKey} = \lambda = 4$$

2. Verschlüsseln $(\mathbb{Z}_{n^2}^*, +) \rightarrow (\mathbb{Z}_{n^2}^*, \circ)$

$$m = 13$$

$$\text{Enc}(m) = g^m \circ r^n \bmod n^2$$

$$r = 23 \quad \left. \right\} \text{Zufallszahl}$$

$$\text{Enc}(13) = 16^{13} \circ 23^{15} \bmod 225 = 47 = c$$

Paillier

②

3. Entschlüsseln

$$c = 71$$

↓
anderes c als vorher

mod inverse
1

$$\text{Decr}(c) = L(c^\lambda \bmod n^2) \circ (L(g^\lambda \bmod n^2))^{-1} \bmod n$$

$$\text{Definition: } L(z) = \frac{z-1}{n}$$

$$\text{Decr}(71) = L(71^4 \bmod 225) \cdot (L(16^4 \bmod 225))^{-1}$$

$$L(c^\lambda \bmod n^2)$$

$$L(71^4 \bmod 225) = L(181) = \frac{181-1}{15} = 12$$

$$(L(g^\lambda \bmod n^2))^{-1}$$

$$L(16^4 \bmod 225) = L(61) = \frac{61-1}{15} = 4 \quad / \text{mod inverse}$$

$$4^{-1} \bmod 15 = 4$$

$$\text{Decr}(71) = (12 * 4) \bmod 15 = 48 \bmod 15 = \underline{\underline{3}}$$

11 SW11 - eVoting

11.1 Aufgabe 2: Wahlprotokoll mit symmetrischer Verschlüsselung

Wir betrachten zwei Wählerinnen Alice und Eve. Am Ende des symmetrischen eVoting Protokolls mit blinder Signatur möchte Eve ermitteln, wie Alice gewählt hat.

Kann Sie es? Begründen Sie Ihre Antwort.

11.1.1 Solution

Nein, Eve kann nicht ermitteln, was Alice gewählt hat. Das wäre nur möglich, wenn Eve im Zählsystem oder als Administrator fungieren würde. So hätte sie die Möglichkeit, die verschlüsselte Stimme von Alice zu rekonstruieren oder aufgrund des von Alice ans Zählsystem übergebenen Schlüssels K die zugehörige Stimme ausfindig zu machen.

Aber da Eve ebenfalls Wählerin ist und sie lediglich die am Ende vom Zählsystem veröffentlichte Liste mit den verschlüsselten Stimmen (samt Schlüssel) sehen kann, hat sie keine Möglichkeit, eine der Stimmen genau Alice zuzuordnen, da sie nicht Alices Schlüssel K kennt.

11.2 Aufgabe 3: additives homomorphes Wahlprotokoll

“Verschlüssle alle Wahlergebnisse (r_i und ‘Gewähltes’ als m) und zähle sie mit dem Paillier Verfahren zusammen (homomorphe Addition). Entschlüssle die Ergebnisse anschliessend und leite von der erhaltenen Zahl das Ergebniss ab.”

Bedeutung: “10=ja” und “1=nein”

Wählerin	Gewähltes	r_i
V1	10	2
V2	1	8
V3	1	4
V4	1	2

Die Verschlüsselung von der Aufgabe 3 (SW10):

$$p = 3$$

$$q = 5$$

$$n = p * q = 15$$

$$g = 16$$

$$\lambda = \text{kgV}((p-1) * (q-1)) = 4.$$

11.2.1 Solution

Paillier II

1. Schlüsselerzeugung

$$n = p \cdot q = 3 \cdot 5 = 15 \quad | \quad g = 16$$

$$n^2 = 15^2 = 225$$

$$\lambda = \text{kgV}(p-1, q-1) = \text{kgV}(2, 4) = 4$$

$$\text{PublicKey } pk = (n, g) = (15, 16)$$

$$\text{PrivateKey} = \lambda = 4$$

2. Verschlüsselung

$$\text{Enc}(m) = g^m \circ r^m \bmod n^2$$

$$c_1 = \text{Enc}(m_1) = \text{Enc}(10) = (16^{10} \cdot 2^s) \bmod 225 = 218$$

$$c_2 = \text{Enc}(m_2) = \text{Enc}(1) = (16^1 \cdot 8^s) \bmod 225 = 137$$

$$c_3 = \text{Enc}(m_3) = 34$$

$$c_4 = \text{Enc}(m_4) = 38$$

$$C_{\text{rest}} = c_1 \cdot c_2 \cdot c_3 \cdot c_4 \bmod n^2 = 218 \cdot 137 \cdot 34 \cdot 38 \bmod 225 = 47$$

3. Entschlüsseln von C_{rest}

$$\text{Decr}(c) = L(c \bmod n^2) \circ (L(g^\lambda \bmod n^2))^{-1} \bmod n$$

$$\text{Decr}(C_{\text{rest}}) = L(47^4 \bmod 225) \cdot ((L(16^4 \bmod 225))^{-1}$$

$$\downarrow \qquad \qquad \downarrow$$

$$\frac{106-1}{15} = 7 \qquad \qquad \frac{61-1}{15} = 4 \Rightarrow 4^{-1} \bmod 15 = 4$$

$$\rightarrow = 7 \cdot 4 \bmod 15 = 13 \Rightarrow 10+1+1+1 \Rightarrow 1x \text{ Ja}, 3x \text{ Nein}$$

$m_1 = 10$	$r_1 = 2$
$m_2 = 1$	$r_2 = 8$
$m_3 = 1$	$r_3 = 4$
$m_4 = 1$	$r_4 = 2$

12 SW12 - ePayment

12.1 Aufgabe 2: Random Serialnumber

Das Protokoll von E-Cash entwickelt von der Firma Dicash sieht eine zufällig gewählte Seriennummer S für jede eMünze vor. Sollte S zufällig gewählt werden oder sollte S wie die Zahl X im Wahlzettel für eVoting von einer bestimmten Struktur sein?

12.1.1 Solution

Die Seriennummer sollte nicht Random sein, da man sonst leicht einen Betrugsversuch machen könnte ala:

- Mallory nimmt generiert eine zufällige Zahl C und verschlüsselt mit einem Public-Key (e, d):
 $S = C^e$
- Mit $C = S^d$ kann die Zahl korrekt entschlüsselt werden und man könnte meinen es sei eine richtige Seriennummer.

Ein Beispiel für eine gute Seriennummer Struktur wäre ein Hash von relevanten Daten der Münze (Issuer, Betrag, Gültigkeitszeitraum)

12.2 Aufgabe 4: Serialnumber Bits

Wie viele Bit muss die zufällig generierte Seriennummer einer E-Münze lang sein, damit die Wahrscheinlichkeit für eine zufällige Übereinstimmung von zwei Nummern kleiner ist als die Wahrscheinlichkeit, bei zwei aufeinander folgenden Ziehungen im Lotto (6 aus 49) sechs Richtige zu

tippen? Tipp: Berechnen Sie zuerst die Wahrscheinlichkeit, mit einer zufällig erzeugten Seriennummer

eine vorgegebene Zahl fester Länge zu treffen. Bestimmen Sie dann deren Länge n .

12.2.1 Solution

Die Wahrscheinlichkeit für sechs Richtige im Lotto bei einem Tipp ist

$$\frac{1}{\binom{49}{6}} = 1/13983816.$$

Die Wahrscheinlichkeit, bei zwei unabhängigen Ziehungen im Lotto jeweils sechs Richtige zu tippen ist

$$\frac{1}{\binom{49}{6}^2} \approx 5 \cdot 10^{-15}.$$

Eine mit einem guten Zufallszahlengenerator erzeugte n -Bit-Zahl hat einen bestimmten vorgegebenen Wert mit Wahrscheinlichkeit $1/2^n$. Diese soll kleiner sein als

$$\frac{1}{\binom{49}{6}^2},$$

also

$$\begin{aligned} 2^n &> \binom{49}{6}^2 \\ \Rightarrow n &> \frac{2 \log \binom{49}{6}}{\log 2} \approx 47,5. \end{aligned}$$

12.3 Aufgabe 5: Secret Splitting

Sei Alices Identitätsinformation $UID = 1000$. Ist die folgende Gleichung korrekt?

$$1000 = 9008 \oplus 8408$$

12.3.1 Solution

Zahlen in Binärsystem umwandeln und XORen:

8192 4096 2048 1024 512 256 128 64 32 16 8 4 2 1

$9008 = 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0$

$8408 = 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 0$

$XOR = 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 0 = 8 + 32 + 64 + 128 + 256 + 512 = 1000$

All good!

13 SW13 - Blockchain

13.1 Aufgabe 1: Blockchain

Erläutern Sie, inwiefern die dezentrale Architektur des Blockchain-Netzwerkes zu dessen Sicherheit beiträgt.

13.1.1 Solution

Aufgrund der architektonischen Struktur bietet die Blockchain eine hohe Sicherheit und eine End-to-End-Transparenz. In einer Blockchain werden Daten anonym gespeichert, dezentrale registriert und übermittelt. Diese Kombination gewährleistet ein hohes Mass an Sicherheit, „die auf sicherer Authentifizierung, Kennzeichnung und Schutz der Daten basiert.“

Eine Manipulation alter Transaktionen ist nicht einfach möglich, dafür müssten alle Hashwerte der Folgeblöcke berechnet werden. Zusätzlich müsste dann noch eine Mehrheit (51%) der Nodes die „neue“ Kette akzeptieren.

13.2 Aufgabe 2: Elektronische Zahlungsverkehr

Woran unterscheidet sich E-Bargeld von Bitcoin? Nennen Sie mindestens 2 Punkte und begründen Sie Ihre Antwort.

13.2.1 Solution

1. Unterschied: Trust

- E-Cash: Durch einen zentralen „General Ledger“ (identisch mit traditionellem Geld).
- Blockchain als Basistechnologie für Bitcoin: Durch eine dezentrale Instanz namens „Distributed Ledger“

2. Unterschied: Eingesetzte Kryptographische Verfahren

- E-Cash: Braucht blinde Signatur (Chaum) einer zentralen Stelle (e.g. Bank). Die zu übertragenden Einheiten werden mit einer Seriennummer versehen und im “central ledger” der zentralen Instanz vermerkt. Asymmetrisches Verfahren zur Generierung von Schlüsseln basiert auf RSA.
- Bitcoin: Verwendet elliptische Kurven und asymmetrische Kryptographie (public key: walletadresse, private key: signieren von Transaktionen).

13.3 Aufgabe 3: Bitcoin

Nennen Sie die verwendeten kryptographischen Elemente für Bitcoin und beschreiben Sie welche Funktionen sie erfüllen?

- a) Erläutern Sie, wie das Problem double-spending für Bitcoin gelöst wird?
- b) Wie beurteilen Sie die ökologischen Aspekte in Bezug auf Energieverbrauch für Mining?
- c) Was passiert im Jahre 2141, wenn kein Bitcoin mehr im Lauf gebracht wird?
- d) Wie sicher ist Bitcoin gegen Cyberattacken? Kann Bitcoin verloren gehen?

13.3.1 Solution

- a.) Das Problem von double-spending wird durch die Arbeitsweise von den Miners und einen Cachespeicher (UTXO), der **Unspent Transaktionsoutputs** der Blockchain gewährleiste. Angenommen eine Absenderin Alice will eine Transaktion durchführen. Der Miner überprüft erst die Verweise auf die Inputtransaktionen (Einzahlungen) von Alice: diese müssen mindestens den Betrag der Outputtransaktionen (Auszahlungen) abdecken. Dann überprüft der Miner über den UTXO, ob die in der Transaktion referenzierten Inputs noch nicht für andere Transaktionen verwendet wurden (wurde das Geld vorher schon ausgegeben-double-Spending?). Sollte das der Fall sein wird die Transaktion abgelehnt.
- b.) Das Mining von Bitcoins verwendet eine sehr hohe Menge an Energie für den POW, was ziemlich unökologisch ist.
- c.) Im Jahr 2141 wird es keine neuen Bitcoins mehr geben für Mining Aktivitäten, dann wird der maximale Betrag von 21'000'000 Bitcoins im Umlauf sien.
- d.) Die Blockchain von Bitcoin ist grundsätzlich Read-Only. Eine Manipulation eines Blocks wird durch das verteilte System schnell aufgedeckt (51% Prinzip). Wenn jemand seinen Private-Key für sein Wallet verliert, dann sind die Bitcoins darin auch verloren (gibt keine Instanz die hier helfen könnte).

14 SW14 - Quanten Kryptografie

14.1 Aufgabe 2: BB84

Suppose Alice uses the following polarization states and bit values and Bob measures in the depicted basis. Compute the raw key (bevor reconciliation/error correction and privacy amplification).

Alice's polar. states	$ \swarrow \rangle$	$ \updownarrow \rangle$	$ \updownarrow \rangle$	$ \leftrightarrow \rangle$	$ \nwarrow \rangle$	$ \nwarrow \rangle$	$ \updownarrow \rangle$	$ \swarrow \rangle$	$ \leftrightarrow \rangle$
Alice's bit value	1	0	0	1	0	0	0	1	1
Bob's basis	\otimes	\otimes	\oplus	\otimes	\oplus	\otimes	\otimes	\oplus	\oplus

14.1.1 Solution

BB84

Alice Polar state	$ \swarrow \rangle$	$ \updownarrow \rangle$	$ \updownarrow \rangle$	$ \leftrightarrow \rangle$	$ \nwarrow \rangle$	$ \nwarrow \rangle$	$ \updownarrow \rangle$	$ \swarrow \rangle$	$ \leftrightarrow \rangle$	$ \updownarrow \rangle$
Alice Bit value	1	0	0	1	0	0	0	1	1	1
Bob's basis	\otimes	\otimes	\oplus	\otimes	\oplus	\otimes	\otimes	\oplus	\oplus	\oplus
Bob receive	\swarrow	\swarrow	\downarrow	\swarrow	\leftrightarrow	\uparrow	\swarrow	\leftrightarrow	\leftrightarrow	\leftrightarrow
Correct	✓	✗	✓	✗	✗	✓	✗	✗	✓	
Secret Key	1	0		0				1		

Key: 1001

14.2 Aufgabe 3: Again BB84

In the BB84 protocol:

1. what is the probability, that Bob chooses the same basis as Alice?

50% (since he only has to choose horizontal or vertical)

2. what is the probability, that Eve guesses the correct basis and resends the qubit in the correct state to Bob? Will her interaction be observed in this case?

Eve has a 50% percent Chance to guess the correct basis.

Eve has another 50% chance to resend the qubit in the correct state. So Eve produces an additional 25% Error Overall

Bob/Alice could detect the listening, because Eve "destroys" the original Qubit with every only has a 75% chance to restore it correctly.

The Probability of Detection is: $p = 1 - 0.75^n$ where n is number of qubits.

3. What percentage of bits of the raw key has Bob to discard typically? Note: the shorter key is called shifted key.

Also 50%

14.3 Aufgabe 4: How to find the period of an injective function f

The idea in classical computing would be to choose random x and x' and check, whether $f(x) = f(x')$.

If that is the case, then $x' = x + kp$ for some $k \in \mathbb{Z}$. Hence we could find a multiple (kp) of the period p .

It can be shown, that a first guess of p can be made using brute force, by finding a collision in f using approximately \sqrt{r} inputs. Explain!

14.3.1 Solution

Das Theorem ist Teil von Shors Algorithmus "find period of a function f".

Es geht also darum zu schauen, ab wann sich eine Funktion wiederholt. Als Beispiel nehmen wir $2^k \bmod 15$

$$\begin{aligned}2^0 \bmod 15 &= 1 \\2^1 \bmod 15 &= 2 \\2^2 \bmod 15 &= 4 \\2^3 \bmod 15 &= 8 \\2^4 \bmod 15 &= 1 \\2^5 \bmod 15 &= 2 \\2^6 \bmod 15 &= 4 \\2^7 \bmod 15 &= 8\end{aligned}$$

Man sieht schnell, dass die Funktion sich nach 4 Schritten wiederholt. somit ist $p = 4$ und es gilt $2^k = 2^{k+p}$

Anscheinend kann man hier das Geburtstags-Paradoxon anwenden und kommt so auf \sqrt{r} Inputs die man testen muss.

14.4 Aufgabe 5: Shor's algorithm to factor 35

Use Shor's algorithm to factor $n = 35$. Start with $m = 11$, and then $m = 2, m = 3$.
For step 2 use a classical computer, i.e. do it by hand!

14.4.1 Solution

Shors Algorithmus

Gegaben: $n = 35$

Vorschlag: $m = 11, n = 2, m = 3$

① Prüfe das: $m < n$ // finde random m
 $\gcd(m, n) = 1$

z.B. $\gcd(11, 35) = 1 \quad \checkmark$

② Periode erraten von: $11^k \bmod 35 \quad (k=0, 1, 2, 3, \dots, \infty)$

$11^0 \bmod 35 = 1$
 $11^1 \bmod 35 = 11$
 $11^2 \bmod 35 = 16$
 $11^3 \bmod 35 = 1$
 $11^4 \bmod 35 = 11$
 $11^5 \bmod 35 = 16$

③ Prüfe ob p ungerade, wenn ja \Rightarrow anderes m wählen

Für $2^k \bmod 35$ finden wir $p=12$

(4) Berechne

$$\left(m^{\frac{p}{2}} - 1\right) \cdot \left(m^{\frac{p}{2}} + 1\right) = m^p - 1 = 0 \bmod n$$

Achtung: Wenn $m^{\frac{p}{2}} + 1 \equiv 0 \pmod{n}$ gilt, zurück zu Schritt 1
 $m^{\frac{12}{2}} + 1 = 2^6 + 1 \bmod 35 = 30 \quad \checkmark$

$$(2^6 - 1) \cdot (2^6 + 1) = 2^6 - 1 = 0 \bmod 35 \quad \checkmark$$

(5) Berechne: $d = \gcd(m^{\frac{p}{2}} - 1, n)$

$$\gcd(2^6 - 1, 35) = \gcd(28, 35) = 7 = d$$

(5.1) Find 'non-trivial factor' of $7 = \gcd(28, 35)$

$$n = p \cdot q \quad // \text{wobei } d = 7 \text{ } p \text{ oder } q \text{ ist}$$

$$\text{Somit: } d \cdot q = n \Rightarrow 7 \cdot q = 35 \Rightarrow \underline{\underline{q = 5}}$$