Cäsar und die modulare Arithmetik - Lösung

1.

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

2. *G U T verschlüsselt mit s=9:* _____*P D C*_____

3.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	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

4. *Verschlüssle nun mit dem folgenden Verfahren:*
Schreibe den Buchstaben als Zahl → addiere den Schlüssel → schreibe als Buchstaben

G	→ 6	→	6 + 9 = 15 → 15
U	→ 20	→	20 + 9 = 29 → ??? D (3)
T	→ 19	→	19 + 9 = 28 → ??? C (2)

5. Die Zahlen werden größer als 25.

6. Rechne mod 26

7. (Klartextbuchstabe + Schlüssel) mod 26 = Kryptobuchstabe

8. (Kryptobuchstabe - Schlüssel) mod 26 = Klartextbuchstabe

(16-9)mod26 = 7 (4-9)mod26 = -5 mod26 = 21 (3-9)mod26 = -6 mod26 = 20

Hinweis: Die Java-Operation % berechnet für negative Zahlen nicht das Gleiche wie mod erwarten lassen würde.

9. *S*timmt weil: (K-S)mod26 = (K+26-S)mod26

10. Im Vigenere Verfahren, im One-Time-Pad, in der Skytale-Verschlüsselung und vielen anderen.

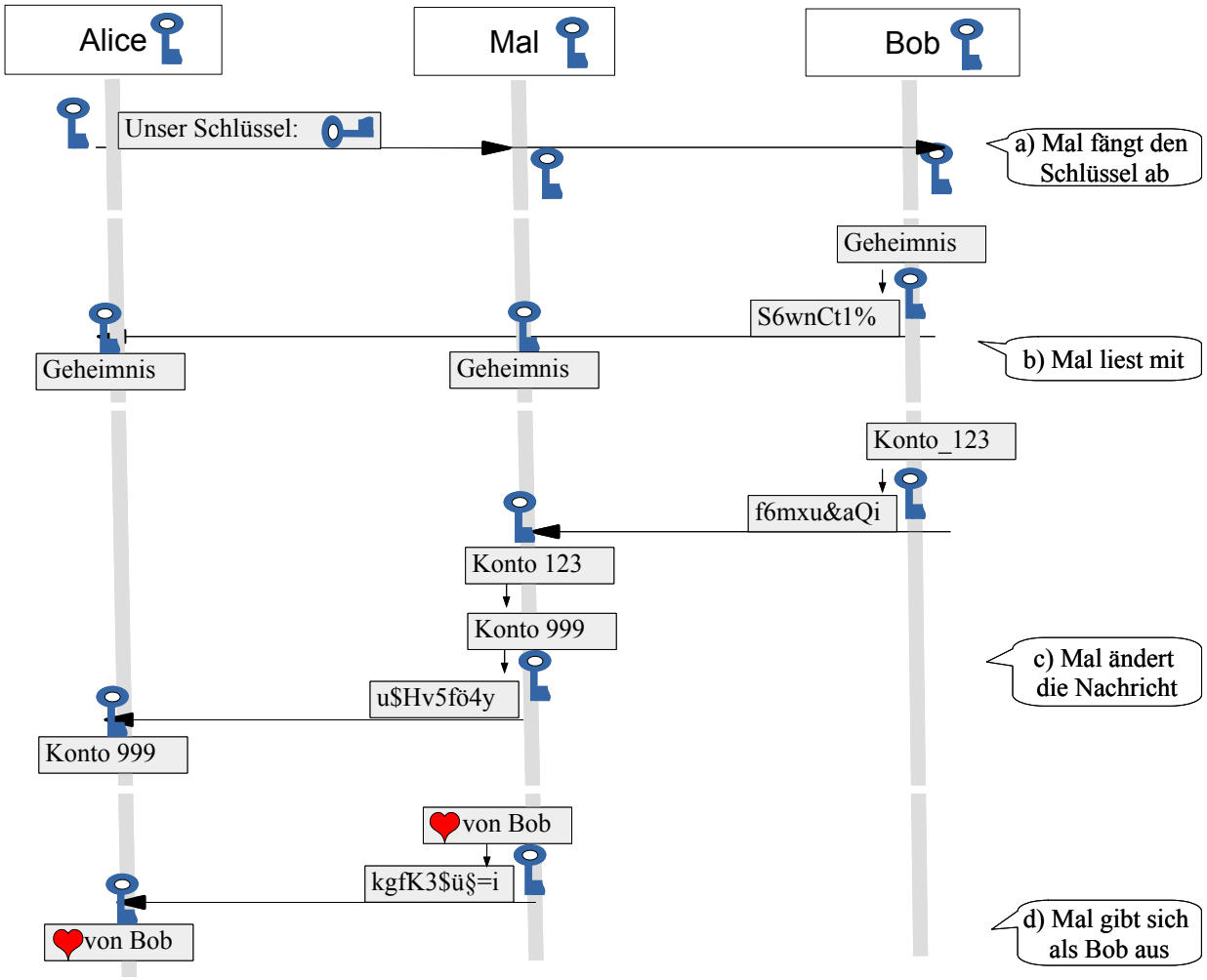
IuD: Kryptologie - Wiederholung - Lösung

Aufgaben:

1. Kryptoverfahren:

- Transpositionsverfahren
  - Skytale: sehr leicht zu brechen, da nur sehr wenige Schlüssel in Frage kommen.
- Substitutionsverfahren:
  - Cäsar: monoalphabetisch, sehr leicht zu brechen (nur 25 mögliche Schlüssel)
  - Allgemeine monoalphabetische Substitution: viele Schlüssel (25!), aber mit Häufigkeitsanalyse zu brechen.
  - Vigenère: polyalphabetisch, schwieriger zu brechen, aber mit zweistufigem Verfahren (Angriff auf Schlüssellänge, danach Häufigkeitsanalyse der Teiltexte) möglich.
  - One-Time-Pad: polyalphabetisch, absolut sicher, Nachteil: Schlüssel so lang wie die Nachricht, nur einmal verwendbar → Problem des Schlüsseltauschs
- Modernes Verfahren:
  - AES: Kombination aus Transposition und Substitution, nicht 100% sicher, aber zur Zeit ein praktikabler Kompromiss.

2.



3.

Angriffsmöglichkeit:	Krypto-Ziel:
Mitlesen der Nachricht	1. Vertraulichkeit
Ändern der Nachricht	2. Integrität
Absender fingieren	3. Authentizität *

* Das Ziel Verbindlichkeit wird hier nicht weiter differenziert.

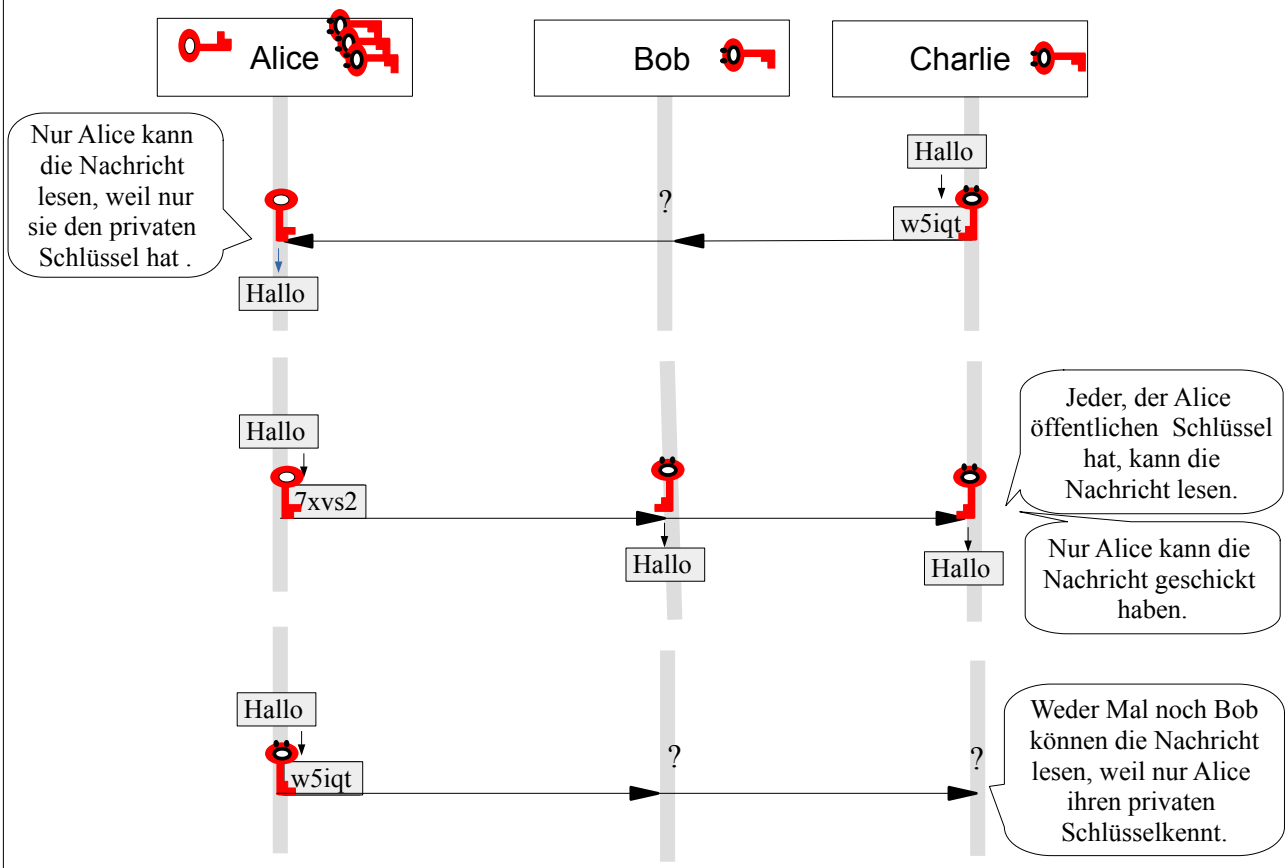
4. Zentrales Problem ist der Schlüsseltausch.

IuD: Asymmetrische Verschlüsselung - Lösung

Aufgaben:

1. Das Ver- und Entschlüssel sind symmetrisch: beide verwenden denselben Schlüssel. Beim asymmetrischen Verfahren sind beide Verfahren asymmetrisch, weil beide verschiedene Schlüssel nutzen.

2.



a) Jeder, der einen öffentlichen Schlüssel von Alice hat, kann ihr geheime Nachrichten schreiben.

b) Alice kann niemandem geheime Nachrichten schreiben.

c) B und C können nicht geheim kommunizieren.

3. a) Nachrichten an A: Krypto-Ziel der Vertraulichkeit.

b) Nachrichten von A zu irgendjemandem: Krypto-Ziel der Authentizität.

Bob kann nicht sicher sein, dass niemand sonst die Nachricht gelesen hat.

Bob kann sicher sein, dass die Nachricht von Alice kommt.

4.

	Asymmetrisch	Symmetrisch
a) Anzahl Schlüssel:	**5** Schlüsselpaare **n** Schlüsselpaare	4+3+2+1= **10** Schlüssel (n-1)+(n-2)+...+1 = **(n-1)·n : 2** Schlüssel (bei n= 100: 4950 Schlüssel)
b) Jeder verwaltet:	das eigene Schlüsselpaar sowie die öffentlichen Schlüssel der anderen, also: 4+2= **6 Schlüssel** n+1 Schlüssel	4 Schlüsssel n-1 Schlüssel

5. Verschlüsseln mit asymmetrischer Verschlüsselung: (Aufgabe mit Chat-Tool)

e) Verwendeter Schlüssel: der öffentliche Schlüssel des Empfängers

f) Verwendeter Schlüssel: mein privater Schlüssel

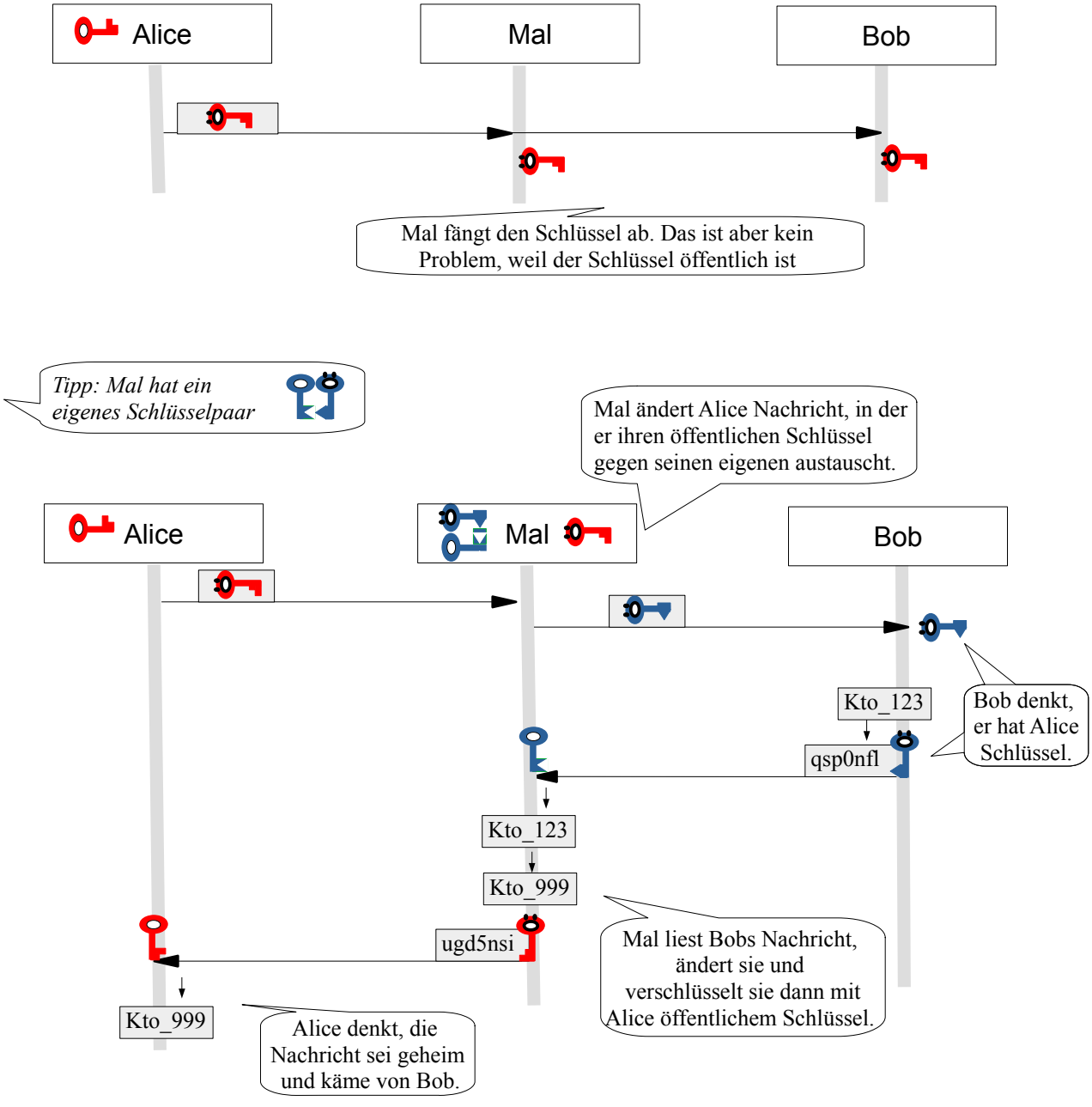
g) Verwendeter Schlüssel: mein privater Schlüssel

h) Verwendeter Schlüssel: der öffentliche Schlüssel des Absenders. Nur so kann ich sicher sein, dass der angegeben Absender die Nachricht geschrieben hat.

IuD: Man-in-the-middle-Angriff - Lösung

Aufgaben:

1.



2. Bobs Problem: Kommt der (öffentliche) Schlüssel wirklich von Alice? Die Authentizität ist nicht sichergestellt.

3. Man-in-the-middle-Angriff - indviduell (Aufgabe mit Chat-Tool)

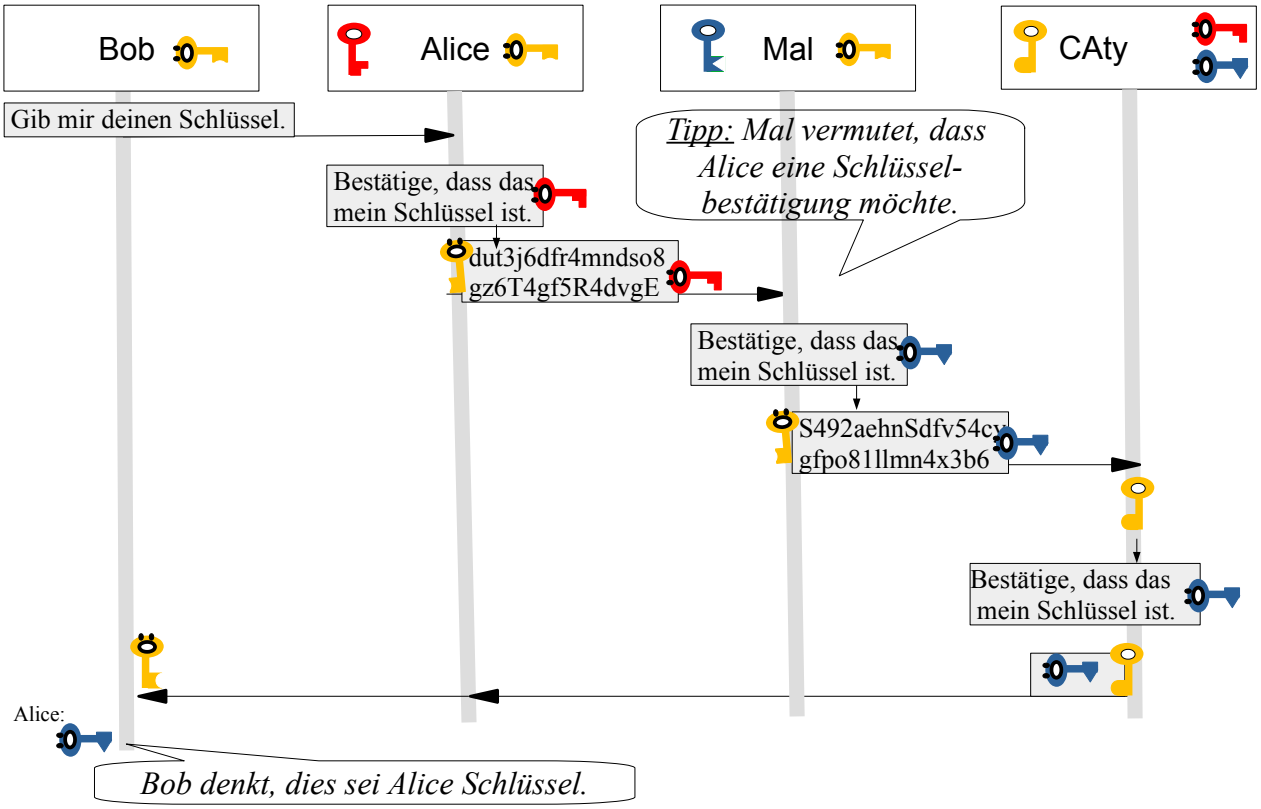
4. Ablauf: Bob fragt Alice nach ihrem öffentlichen Schlüssel. Alice bittet CAty zu bestätigen, dass es sich um Alice Schlüssel handelt. CAty wendet auf die Nachricht mit Alice Schlüssel ihren eigenen privaten Schlüssel an und schickt sie zu Bob. Bob wendet Catys öffentlichen Schlüssel an, verifiziert also die Nachricht und hat Alice öffentlichen Schlüssel.

a) Nein, nur CAty kann sie lesen, weil Alice sie mit CAtys öffentlichen Schlüssel verschlüsselt hat. Die Vertraulichkeit der Nachricht ist sichergestellt.

b) Mal kann die Nachricht, die Alice Schlüssel beinhaltet, lesen, weil er CAtys öffentlichen Schlüssel hat. Das ist aber nicht problematisch, da es sich um den öffentlichen Schlüssel von Alice handelt, den jeder haben darf.

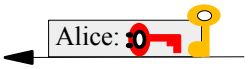
c) CAty wendet auf die Nachricht, die Alice Schlüssel beinhaltet, ihrem eigenen privaten Schlüssel an, um zu kennzeichnen, dass die Nachricht tatsächlich von CAty kommt. (Authentizität). Damit kann Bob sicher sein, dass die Nachricht von CAty stammt.

d) Nein, es ist nicht sicher:

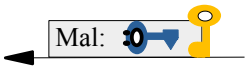


Mal ändert Alice Nachricht und bittet um Bestätigung seines eigenen Schlüssels. Bob erhält eine Antwort, „verschlüsselt“ mit CAtys privatem Schlüssel. (Die Nachricht stammt ja auch von CAty, sie ist aber nicht die Antwort auf Bobs, bzw. Alice Frage.) Er ist sich nun sicher, dass er Alice Schlüssel hat. Nun kann Mal Bobs Nachrichten an Alice lesen und ändern. Weiterhin kann Mal in Bobs Namen Nachrichten an Alice schicken.

e) Statt nur den Schlüssel zu verschicken, wird der Name, also Alice, mitgeschickt.



Fordert Mal von CAty eine Schlüsselbestätigung und leitet diese an Bob weiter, so würde Bob die Vertauschung merken.



IuD: Zertifikat - Lösung

Aufgaben

1. Sicherer Schlüsselaustausch (Aufgabe mit Chat-Tool)

Nein er hat keine Chance.

IuD: Digitale Signatur - Lösung

Aufgaben:

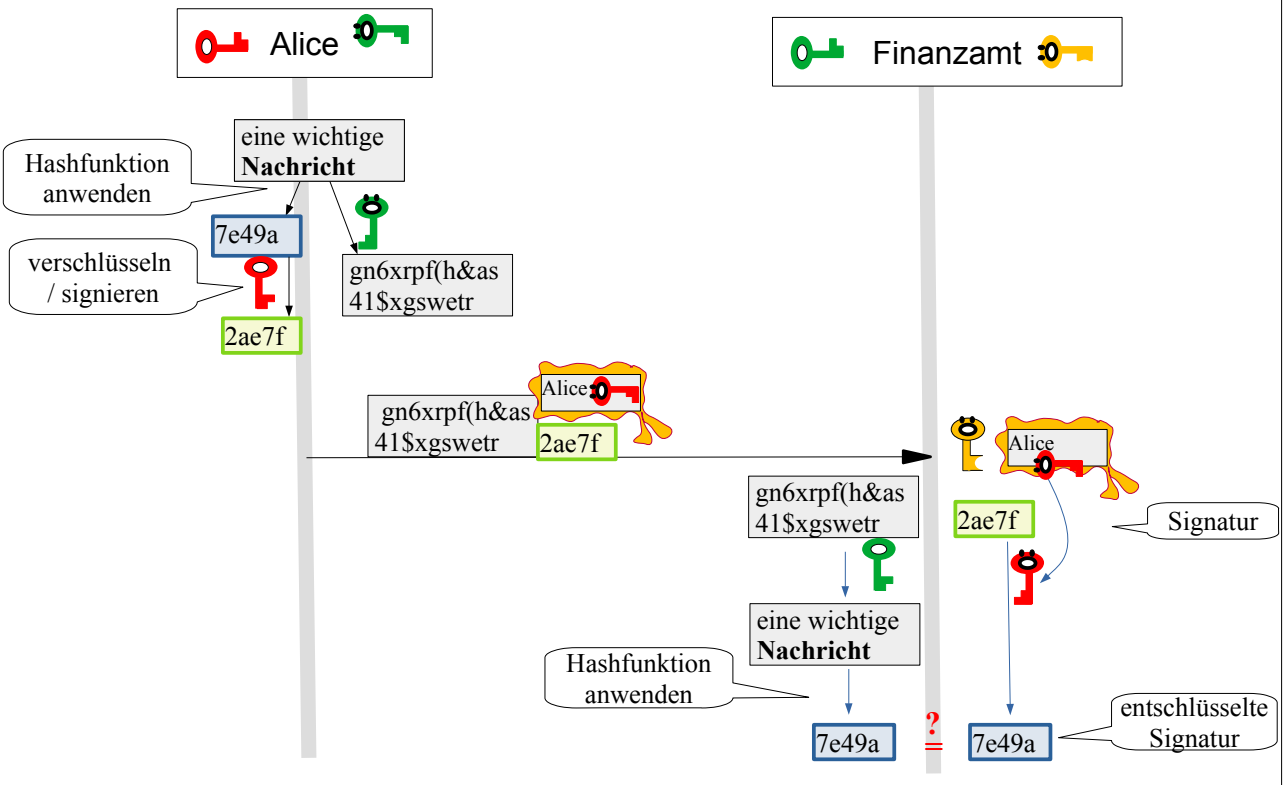
1. a) Es wird nicht die komplette Nachricht verschlüsselt, sondern nur ein kleiner Teil, der Fingerabdruck der Nachricht. Dazu wird mit einer sogenannten Hash-Funktion ein Hash-Wert berechnet. Das ist der Fingerabdruck der Nachricht. Nur auf diesen (viel kleineren) Hash-Wert wird der privaten Schlüssel angewendet. Das Ergebnis ist die Signatur. Die Signatur wird zusammen mit der Nachricht an den Empfänger geschickt. Das Zertifikat mit dem öffentlichen Schlüssel schickt Alice gleich mit. Der Empfänger trennt die Nachricht von der Signatur. Auf die Nachricht wendet er die Hash-Funktion an und erzeugt den Fingerabdruck der Nachricht. Parallel dazu wendet er auf die Signatur den öffentlichen Schlüssel aus dem Zertifikat an und erhält den Fingerabdruck der Nachricht. Diese beiden Fingerabdrücke vergleicht er. Sind sie gleich, stammt die Nachricht tatsächlich von Alice.

b) Die Nachricht stammt tatsächlich von Alice, wenn der Hashwert der Nachricht und die entschlüsselte Signatur gleich sind.

c) Wenn Mal entweder die Nachricht ändert oder die Signatur, dann sind Hashwert der Nachricht und entschlüsselte Signatur nicht gleich.

d) s.u.

e) Signaturen werden überall dort verwendet, wo eine elektronische Unterschrift eine Unterschrift auf Papier ersetzen soll. Z.B. Anträge bei Behörden, Steuererklärung beim Finanzamt, Firmen, die anderen Firmen Rechnungen stellen,… Aber auch um z.B. Dateien fälschungssicher aufzubewahren.



2. Signieren einer Nachricht: (Aufgabe mit Chat-Tool)

a) Verwendeter Schlüssel: mein privater Schlüssel

b) Verwendeter Schlüssel: öffentlicher Schlüssel des Senders

c) Nein, das ist nicht möglich.

3. Versenden eine verschlüsselten und signierten Nachricht: (Aufgabe mit Chat-Tool)

Signatur erzeugen und mit meinem privaten Schlüssel verschlüsseln.

Die Nachricht mit dem öffentlichen Schlüssel des Empfängers verschlüsseln.

4. Nein er hat keine Chance. (Aufgabe mit Chat-Tool)

5. a) Alice stellt die Anfrage an Bobs Internetseite, dass sie kommunizieren will. Bob sendet ihr sein Zertifikat. Alice Browser verifiziert das indem er den öffentlichen Schlüssel der CA anwendet und merkt sich Bobs öffentlichen Schlüssel. Es wird eine Nachricht an Bob geschickt mit einem einfachen symmetrischen Schlüssel. Die folgende Kommunikation findet mit diesem symmetrischen Schlüssel statt. Wenn die Seite verlassen wird , wird der Schlüssel wieder gelöscht.

b) Die Verschlüsselung mit einem symmetrischen Schlüssel ist viel schneller als mit dem asymmetrischen Schlüssel.

c) Ja, weil ein Angreifer den symmetrische Schlüssel zwar theoretisch mit brute force knacken kann, aber praktisch nicht genug Zeit dazu hat.

6. a) Wesentlichen Inhalte eines Zertifikats sind z.B. :

Inhaber: ...
Aussteller: ...
Gültigkeit von … bis ...
öff. Schlüssel: Algorithmus: ...RSA
Schlüssellänge: ...2048
Exponent: ...65537
Modulus: ...B1:89:9E:….
SerienNr: ...
Fingerabdrücke: ...SHA-256: 82:A7:0D:58:….
Schlüsselverwendung: ...digitale Signatur, key Encrypment
...

b) Die gängigen Browser beinhalten diesen öffentlichen Schlüssel und zeigen das Zertifikat bereits „entschlüsselt“ also verifiziert an. Weiterhin ist das Zertifikat kein Geheimnis, sondern für jeden zugänglich. Die Verschlüsselung dient nur der Authentifizierung und soll sicherstellen, dass es tatsächlich von der Zertifizierungsstelle ausgestellt wurde.

c) Ein Fingerabdruck ist ein Text, der viel kürzer ist als die eigentlichen Nachricht und die Nachricht eindeutig „repräsentiert“. Aus dem Fingerabdruck kann man die ursprüngliche Nachricht (praktisch) nicht wieder herleiten. Das ist ähnlich einem menschlichen Fingerabdruck. SHA-256 ist der Name eines standardisierten Hashalgorithmus (Hashfunktion), mit dem man einen solchen Fingerabdruck erzeugen kann.

7. Es ist nicht sicher, ob das Zertifikat tatsächlich vom angegebenen Absender stammt. Es

könnte auch abgelaufen sein (‚gültig bis' wurde überschritten).

Es könnte also sein, dass der Eigentümer versäumt hat, sein Zertifikat verlängern zu lassen, dann würde zunächst keine Gefahr bestehen. Es könnte sich aber auch um ein gefälschtes Zertifikat handeln. Akzeptiert man es, ist jegliche Aktivität, die auf diesem Zertifikat beruht, nicht nur nicht sicher, sondern täuscht eine falsche Sicherheit vor.

8. Zertifikate werden auch von kommerziellen Stellen ausgegeben. Man kennt die Zertifizierungsstelle oft nicht, muss ihr aber vertrauen. Relevante Fragestellungen sind:

Welche Verfahren wurden bei der Ausstellung verwendet? Wie sicher ist das (für meine Anwendung)?

Wie genau prüft die Zertifizierungsstelle die Identität des Zertifikatseigentümers?

Wenn ein Zertifikat gesperrt wird, dann muss diese Sperrinformation an alle gelangen, die das Zertifikat verwenden könnten. Wer aktualisiert/prüft Sperrlisten?

u.s.w. Siehe dazu auch [1].

9. Hashfunktion

- Der Hashwert ändert sich wesentlich, unabhängig davon, ob nur ein einzelnes Zeichen geändert wird oder der ganze Text. Der Hashwert ist immer gleich lang.
- Weil der Hashwert kürzer sein kann als der Ursprungstext, muss es Hashwerte mit mehreren Ausgangstexten geben.
- Es ist quasi unmöglich, zu einem Hashwert einen Ausgangstext zu erfinden.

10. Der SHA-2, *secure hash algorithm*, wird aktuell zu Verwendung empfohlen.[2]

11. Ein PGP-System ist ein hybrides System, bei dem die Nachricht mit einem zufällig erzeugten (symmetrischen) Schlüssel verschlüsselt wird. Nur dieser Schlüssel wird asymmetrisch verschlüsselt und zusammen mit der Nachricht verschickt. Vorteil: schneller, weniger aufwendig. PGP-Syteme basieren häufig auf dem Web of Trust. (siehe z.B. Wikipedia [3])

12. Bob hat Alice Schlüssel und hat sichergestellt, dass es wirklich der Schlüssel von Alice ist. Das kennzeichnet er auf dem Schlüsselserver. Carl möchte nun Alice Schlüssel haben, kennt sie aber nicht. Er sieht auf dem Schlüsselserver, dass Bob den Schlüssel von Alice als echt gekennzeichnet hat. Da es Bob kennt und vertraut, vertraut er auch Alice Schlüssel. So entsteht ein Netzwerk von Vertrauensbeziehungen. (Siehe z.B. inf-schule [4] oder genauer: Wikipedia [5])

1 *https://de.wikipedia.org/wiki/Public-Key-Zertifikat (abgerufen, 27.3.20)*

2 *https://de.wikipedia.org/wiki/SHA-2, (abgerufen 03.01.2020)*

3 *https://de.wikipedia.org/wiki/Pretty_Good_Privacy (28.3.20)*

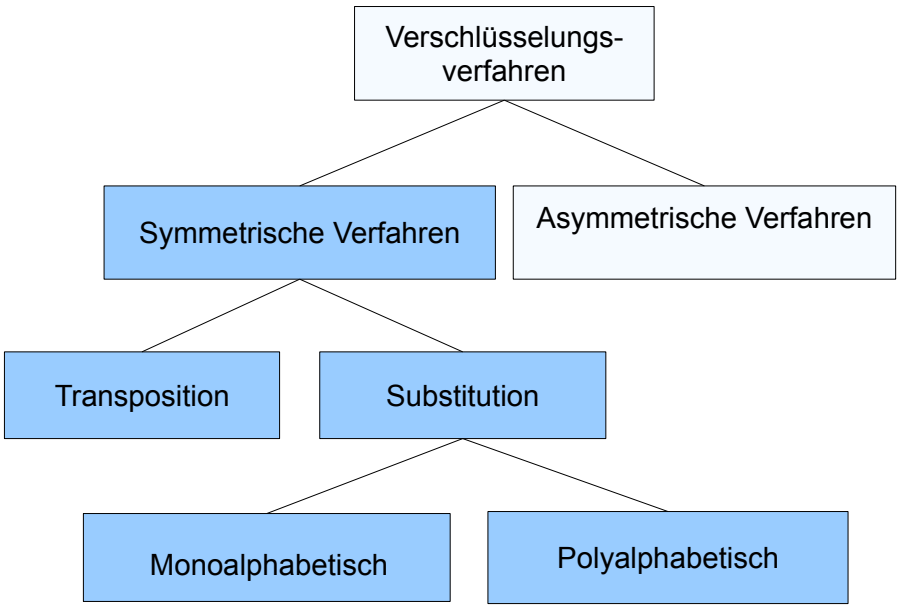
4 *https://www.inf-schule.de/kommunikation/kryptologie/sicherheitsinfrastruktur/konzept_weboftrust (28.3.20)*

5 *https://de.wikipedia.org/wiki/Web_of_Trust (28.3.20)*

Vergleich der Krypto-Verfahren - Lösung

Aufgaben:

1. WDH aus Klasse 8:



Symmetrische Verfahren: Ver- und Entschlüssler haben den gleichen Schlüssel. Die Nachricht wird mit dem gleichen Schlüssel ver- und entschlüsselt.
Bsp.: AES, alle unten stehenden

Asymmetrische Verfahren: Sender und Empfänger haben verschiedene Schlüssel.
Bsp.: RSA

Transpositionsverfahren: Die Buchstaben des Klartextes werden durch die Verschlüsselung anders angeordnet.
Bsp.: Skytale, (Fleißner, Gartenzaun)

Substitutionsverfahren: Die Buchstaben des Klartextes werden bei der Verschlüsselung durch andere Buchstaben (oder Zeichen) ersetzt.
Bsp.: Caesar, Substitutionschiffre, Vigenère, One-Time-Pad

Monoalphabetische Substitution: ein Substitutionsverfahren, bei dem nur ein einziges Schlüsselalphabet verwendet wird.
Bsp.: Caesar, Substitutionschiffre, alle Varianten von Cäsar, RSA (wenn man es fälschlicherweise auf einzelne Buchstaben anwendet).

Polyalphabetische Substitution: ein Substitutionsverfahren, bei dem mehrere Schlüsselalphabete verwendet werden.
Bsp.: Vigenère, One-Time-Pad

2.

Verfahren	(Wie) kann es gebrochen werden? z.B. Anzahl der Möglichkeiten bei brute force etc.	Bewertung: wie sicher ist das Verfahren?
Skytale	Brute force: n Möglichkeiten (bei einer Nachricht der Länge n)	sehr unsicher
Cäsar-Verfahren	Brute force: 25 Möglichkeiten Oder: Häufigkeitsanalyse (1 Buchstabe genügt)	sehr unsicher
Cäsar mit zufälliger Anordnung der Buchstaben	Brute force: (n-1)! Möglichkeiten (bei einer Nachricht der Länge n) Aber leicht zu brechen mit: Häufigkeitsanalyse (über alle Buchstaben)	(sehr) unsicher
Vigenère-Verfahren	a) Schlüssellänge ermitteln: z.B.Kasiski Test b) Je Gruppe: Häufigkeitsanalyse (1 Buchstabe genügt)	unsicher
One-Time-Pad OTP	Absolut sicher! Selbst mit Brute force nicht zu brechen. Aber: Da der Schlüssel genauso lang sein muss wie die Nachricht, nur einmal verwendet werden darf, und auf einem sicheren Weg übermittelt werden muss, dann kann man über diesen Weg auch gleich die Nachricht selber übermitteln.	Sehr sicher, aber unpraktikabel
AES	Mathematisch ist AES mit brute force zu brechen, aber wenn der Schlüssel entsprechend lang ist, ist es nicht in vertretbarem Zeitaufwand machbar (zumindest mit den zur Zeit zur Verfügung stehenden Rechnern). Problem: Schlüsseltausch	Zur Zeit sicher, bis auf Schlüsseltausch
RSA	Wählt man ausreichend große Zahlen, dann lassen sie sich nicht in einer akzeptablen Zeit in ihre Primfaktoren zerlegen (zumindest mit den zur Zeit zur Verfügung stehenden Rechnern). Nachteil: Asymmetrische Verschlüsselung ist viel langsamer als symmetrische. Sinnvoll also nur bei kleinen Datenmengen. Problematisch: Vertrauen in Zertifizierungsinstanzen nötig	Zur Zeit sicher

Hybrid-verfahren	Mittels RSA wird ein (zufälliger) symmetrischer Schlüssel getauscht (→ langsam, aber nur kleines Datenvolumen). Danach wird mit dem symmetrischen Schlüssel verschlüsselt - z.B. mittels AES.	Zur Zeit sicher

3. a) 2000 Bit (vom BSI empfohlen bis zum Jahr 2022, danach 3000 Bit)

b) Mindestens 128 Bit (z.B. AES-128, AES-192, AES-256)

Quelle: BSI[6]

6 *https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102.pdf;jsessionid=4D2998F7DAF2C44ED7836F2D7164AA1E.1_cid503?__blob=publicationFile&v=12* (abgerufen 14.5.20)