Kryptologie

Daten verschlüsseln · Schlüsselaustausch · Asymmetrische Kryptosysteme · Identitätsbeweis

FÜR DEN INFORMATIKUNTERRICHT AM GYMNASIUM

INF school

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Ausgabe Oktober 2026

Florian Köhl

Inhaltsverzeichnis

1 **Einleitung und Terminologie** 5
Historische Entwicklung und Bedeutung 6
Grundbegriffe 7

2 **Geheimschriften** 9
Atbash 11
Lateinischer Polybios 11
Geheimschrift der Freimaurer 13

3 **Kryptosysteme** 15
Das Kryptosystem Caesar 17
Monoalphabetische Substitution und ihre Kryptoanalyse 20
Polyalphabetische Kryptosysteme 29
Maximale Sicherheit: Das One-Time-Pad-Verfahren 31

4 **Enigma** 35
Aufbau und Funktionsweise 36
Entschlüsselung und Schlüsselmenge 39
Cracking Enigma: Turing und die *Bombe* 42

5 **Binäre Kryptosysteme** 48
Binäres One-Time-Pad 49
Stream Ciphers 54
Block Ciphers 56

6 **Schlüsselaustausch** 65
Zwei Schlösser, eine Truhe 65
Einwegfunktionen 70
Der Diffie-Hellman-Schlüsselaustausch 74
Man-in-the-middle-Attacke 77

7 **Public-Key-Kryptosysteme** 80
Public-Key-Kryptologie für Verschlüsselungszwecke 82
Public-Key-Kryptologie für Authentifizierungszwecke 84

8 **(Kryptologische) Hashfunktionen** 90
Hashkollisionen 92

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Kryptologische Hashfunktionen 92
Hashfunktionen als Prüfsumme 93
Digitale Signaturen 94

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Abbildung 3.4: Skytale mit vier Seiten: Der Klartext wird zeilenweise auf den Stab geschrieben, das abgewickelte Band enthält den Geheimtext.

AUFGABE 7: SCHLÜSSEL DER SKYTALE

(a) Was entspricht dem Schlüssel im Kryptosystem der Skytale?
(b) Wieviele mögliche Schlüssel gibt es zu einem Klartext der Länge m?

Monoalphabetische Substitution und ihre Kryptoanalyse

Die Geheimschrift der Freimaurer, das Polybios-Verfahren und das Kryptosystem Caesar gehören alle zu den **Substitutionsverfahren**: ein Klartextbuchstabe wird durch einen Buchstaben aus dem Geheimtextalphabet ersetzt oder substituiert.

Wenn jedes Symbol im Klartext immer durch das gleiche Symbol im Geheimtext ersetzt wird, so nennt man dies **monoalphabetische Substitution**[1] Bei der Geheimschrift der Freimaurer wird der Buchstabe A immer mit dem gleichen Symbol ersetzt, bei Caesar wird bei einem gegebenen Schlüssel das A immer auf den gleichen entsprechenden Buchstaben abgebildet.

Nun wollen wir uns um den kryptoanalytischen Aspekt kümmern und untersuchen, wie und vor allem wie einfach wir diese Systeme knacken können. Wir gehen hier davon aus, dass wir den ursprünglichen Klartext nicht kennen, diesen aber als solchen erkennen, wenn wir auf welche Art und Weise auch immer darauf stossen.

[1] *Mono* ist griechisch und bedeutet so viel wie "eines". Damit drücken wir aus, dass wir bei monoalphabetischen Substituionen genau ein Alphabet für die Ersetzung verwenden.

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

AUFGABE 8: EINFACHE KRYPTOSYSTEME KNACKEN

(a) Wie würdest du einen Geheimtext knacken, der mit dem Caesar-Verfahren verschlüsselt wurde?
(b) Wie würdest du einen Geheimtext knacken, der mit dem Skytale-Verfahren verschlüsselt wurde?
(c) Überleg dir für beide Vorgehensweisen, wie viel Male du jeweils maximal "raten" müsstest.

Die wohl simpelste Methode, ein Kryptosystem zu knacken, ist die so genannte **Brute Force-Methode**. Dabei werden einfach alle möglichen Schlüssel durchprobiert, bis der richtige gefunden wird. Je mehr mögliche Schlüssel ein Kryptosystem hat, desto resistenter ist es folglich gegen diesen naiven Ansatz, da mehr Schlüssel durchgetestet werden müssen.

BEISPIEL: CAESAR KNACKEN[2]

Im Kryptosystem Caesar gibt es 26 mögliche Schlüssel. Einer davon (welcher?) ist aus offensichtlichen Gründen eher unsinnig, wir müssen ihn aber trotzdem in die Analyse einbeziehen. Um einen Geheimtext, der mit Caesar verschlüsselt wurde, zu knacken, müssen wir also lediglich 26 Möglichkeiten durchtesten. Oft reicht es auch, nur einzelne Wörter oder Abschnitte zu testen, um noch mehr Zeit zu sparen.
Dafür brauchen wir nicht mal einen Computer, das kann man auch von Hand innert kürzester Zeit knacken. Es empfiehlt sich also keinesfalls, streng vertrauliche Informationen mit diesem Verfahren zu verschlüsseln!

AUFGABE 9: DOPPELT GEMOPPELT

Paul verschlüsselt sein streng geheimes Tagebuch mit dem Caesar-Verfahren. Er verschlüsselt die Texte gleich doppelt: Nachdem er die Texte einmal verschlüsselt hat, verschlüsselt er den so entstandenen Geheimtext gleich nochmals, mit einem zweiten Schlüssel.

(a) Wird es dadurch schwieriger, das Tagebuch von Paul zu knacken?
(b) Wie sieht es aus, wenn Paul für die zweite Verschlüsselung das Skytale-Verfahren verwenden würde?

Wir wollen in diesem Abschnitt ein Kryptosystem entwickeln, das mit der Brute-Force-Methode nicht so einfach zu knacken ist. Anschliessend erarbeiten wir gleich ein Verfahren, mit dem sich das System trotzdem knacken lässt. Eine grosse Schlüsselmenge allein macht ein Kryptosystem allerdings noch nicht automatisch sicher. Sie schützt zunächst einmal nur gegen den naiven Angriff, wirklich alle Schlüssel durchzuprobieren.

Die grösste Schwäche des Caesar-Systems ist die kleine Anzahl an möglichen Schlüsseln. Wir wollen vorerst beim Ansatz der Substitution bleiben, jedoch die Menge an Schlüsseln deutlich erhöhen. Dazu betrachten wir zuerst nochmals die von Caesar verwendete Substitutions-

[2] Et tu, brute (force)?

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

methode und stellen die Zuordnung der Klartextbuchstaben zu den Geheimtextbuchstaben graphisch dar.

Abbildung 3.5: Caesar mit Schlüssel 3

Wir können die Zuordnung aber auch völlig willkürlich vornehmen! Wir müssen lediglich sicherstellen, dass jeder Buchstabe des Klartextalphabetes mit genau einem Buchstaben aus dem Geheimtextalphabet verknüpft ist. Eine Möglichkeit für eine solche Zuordnung ist in Abbildung 3.6 dargestellt.

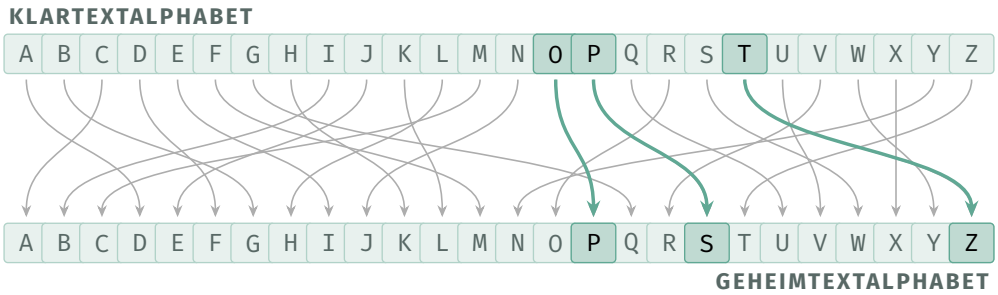

Abbildung 3.6: Willkürliche Zuordnung

Daraus können wir nun ein neues Kryptosystem ableiten:

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

DAS KRYPTOSYSTEM LATPERM

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

Abbildung 3.7: Klartextalphabet (oben) und Permutation (unten), abgeleitet aus Abbildung 3.6

Klartextalphabet: Lateinisches Alphabet

Geheimtextalphabet: Lateinisches Alphabet

Schlüssel: Alle Permutationen des Klartextalphabets

Verschlüsselung: Wähle als Schlüssel eine Permutation des Klartextalphabetes und notiere diese unterhalb des regulären Klartextalphabetes. Für jeden Buchstaben im Klartext bestimmen wir dessen Position im Klartextalphabet und ersetzen ihn mit demjenigen, der an der gleichen Position in der Schlüsselpermutation steht.

Entschlüsselung: Für jeden Buchstaben im Geheimtext bestimmen wir dessen Position in der Schlüsselpermutation und ersetzen ihn mit demjenigen, der an der gleichen Position im Klartextalphabet steht.

BEISPIEL: LATPERM

Und wieder verschlüsseln wir den Klartext `TOP SECRET` jetzt mit **LATPERM**. Wir wählen dazu als Schlüssel die Permutation aus Abbildung 3.6. Der erste Buchstabe T steht an 20. Stelle im Klartextalphabet. Wir ersetzen ihn also durch den 20. Buchstaben aus unserem Schlüssel, dem Buchstaben Z. O wird zu P, P zu S und so weiter. Schlussendlich erhalten wir den Geheimtext `ZPS WIAOIZ`.

Ist LATPERM sicherer als Caesar? Wenn wir es wiederum mit der Brute-Force-Methode knacken möchten, müssen wir alle Schlüssel durchprobieren. Wie viele solche Schlüssel gibt es?

AUFGABE 10: SCHLÜSSEL ZÄHLEN UND LATPERM KNACKEN

(a) Wie viele verschiedene Schlüssel gibt es für das LATPERM-Kryptosystem? Überleg dir dazu, wie viele Möglichkeiten du für die Abbildung des ersten Buchstabens (A) hast. Wie viele bleiben danach für den Buchstaben B? Und für C? Wie kombinierst du diese Möglichkeiten dann, um die totale Anzahl Schlüssel zu erhalten?

(b) Kannst man alle Schlüssel von Hand durchprobieren?

(c) Findest du einen besseren Ansatz für die Kryptoanalyse von LATPERM, bei dem man nicht naiv einfach alle möglichen Schlüssel ausprobiert?

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Jede Permutation des Klartextalphabets ist ein Schlüssel, und für ein Alphabet mit n Buchstaben gibt es $n!$ verschiedene Permutationen[3] Wir verwenden für LATPERM das lateinische Alphabet, welches 26 Buchstaben hat. Die Anzahl Schlüssel von LATPERM ist also $26! \approx 4.033 \cdot 10^{26}$. Ziemlich viele!

So viele Schlüssel können wir unmöglich von Hand durchprobieren[4]. Wir brauchen also einen neuen Plan, um LATPERM zu knacken! Wir wollen diesen Plan anhand von folgendem Geheimtext entwickeln, der aus einem deutschen Klartext abgeleitet wurde:

```
ORIO YHONFO TEQSOQJO YRJ SOI HOUOQYHOJRDXJOI BEQJOI ORIOI MEOBOI CHY
CBORFKYLT POQKHX: ”RNP THOQNPJO SRNP IRNPJ, SH DQEXXOX HIDOPOHOQ”,
QROT XRO RPY CH, ”BORM SH DKQ FORIO WEQCHODO WEQ YRQ PKXJ; ESOQ
IOIIO XRO YRQ, BOII SH XEMNPO CH PKUOI DMKHUXJ; OJBK SRO, SKXX SH
SORIOI QKHU YRJ FQKMMOI COQQORXXOXJ HIS YRJ CKOPIOI COQYKMYOXJ? GOSOX
KISOQO TORDO JROQ, BOII OX YRJ ORIOY JKLTOQI FKOYLTJ, JHJ SKXXOMUO,
OX UORXXJ HIS FQKJCJ. SH XEMMXJ KUOQ OYLTRISOI, SKXX RNP XJKOQFOQ URI
KMX SH!”YRJ SROXOI BEQJOI TMED XRO RI ORIOX XORIOQ IKXOIMEONPOQ HIS
XJKNP RPI XE XOPQ, SKXX OQ XRNP WEQ XNPYOQC XOMUXJ COQTMORXNPJO HIS
XRNP THOQ HOUOQBHISOI OQFMKOQJO.

XJEMC KHT SROXOI XROD TMED SRO YHONFO SKWEI, HY RPI KMMOQ BOMJ
KHXCHLEXKHIOI, HOUOQXKP KUOQ SKX DOBOUO ORIOQ XLRIIO HIS WOQTRID
XRNP RI SOYXOMUOI. DROQRD HYKQYJO SRO XLRIIO XRO HIS XED RPQ SKX
POMSOIUMHJ KHX. XJOQUOIS OYLTKIS SRO YHONFO RPQO IRNPJRDFORJ, RISOY
XRO, SRO UOXRODOQRI SOX MEOBOI, ORIOY XE WOQKONPJMRNPOI JROQO, ORIOQ
XLRIIO, OQMRODOI YHXXJO.
```

Zunächst sieht dieser Text sehr wirr aus. Doch beim genaueren Betrachten erkennen wir, dass bestimmte Buchstaben viel öfter vorkommen als andere. So zählen wir beispielsweise 157 mal den Buchstaben O, G hingegen kommt nur ein einziges Mal vor! Wir können ausnutzen, dass in der deutschen Sprache gewisse Buchstaben deutlich mehr benutzt werden als andere! Betrachte dazu die beiden Tabellen auf der folgenden Seite. Tabelle 3.1 gibt die relativen Häufigkeiten der Buchstaben in der deutschen Sprache an, Tabelle 3.2 die Anzahl Vorkommnisse in unserem Geheimtext.

[3] Für den ersten Buchstaben haben wir n mögliche Geheimzeichen, für den zweiten noch $n-1$, für den dritten $n-2$ usw. Gesamthaft also $n \cdot (n-1) \cdot (n-2) \cdot \ldots \cdot 2 = n!$ mögliche Schlüssel.

[4] Selbst wenn wir pro Sekunde eine Permutation probieren könnten, bräuchten wir immer noch 12 Trillionen Jahre, um alle Schlüssel zu testen!

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Tabelle 3.1: Relative Häufigkeit der Buchstaben in deutschen Texten

BUCHSTABE	HÄUFIGKEIT (%)
E	17.4
N	9.78
I	7.55
S	7.27
R	7.00
A	6.51
T	6.15
D	5.08
H	4.76
U	4.35
L	3.44
C	3.06
G	3.01
M	2.53
O	2.51
B	1.89
W	1.89
F	1.66
K	1.21
Z	1.13
P	0.79
V	0.67
J	0.27
Y	0.04
X	0.03
Q	0.02

Tabelle 3.2: Anzahl Vorkommnisse der Buchstaben in unserem Geheimtext

BUCHSTABE	ANZAHL
A	0
B	12
C	13
D	18
E	21
F	11
G	1
H	41
I	71
J	41
K	38
L	9
M	27
N	19
O	157
P	29
Q	54
R	70
S	41
T	15
U	17
V	0
W	6
X	68
Y	28
Z	0

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

4 Enigma

Das faszinierende Kryptosystem Enigma ist von so grosser historischer Bedeutung, dass wir ihm gerne ein eigenes kurzes Kapitel widmen.

Es handelt sich dabei um eine mechanische Verschlüsselungsmaschine, welche auf den ersten Blick einer alten Schreibmaschine ähnelt. Kurz nach dem ersten Weltkrieg vom deutschen Ingenieur Arthur Scherbius erfunden und patentiert, wurde dieses Kryptosystem vor allem durch seine Rolle im zweiten Weltkrieg bekannt. Dort wurde es auf Seiten der deutschen Wehrmacht verwendet, um geheime Botschaften, Anweisungen oder Pläne sicher zu verschlüsseln.

Abbildung 4.1: Eine Enigma. Von vorne nach hinten: Steckerbrett, Tastatureingabe, Lampenfeld, Walzen
Bildnachweis: Wikimedia Commons (Urheber: Alessandro Nassiri), CC BY-SA 4.0

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Aufbau und Funktionsweise

Die für uns wichtigsten Komponenten einer Enigma sind auf Abbildung 4.1 gut zu erkennen. Drückt man einen (Klartext)buchstaben auf der Tastatur, so leuchtet auf dem Lampenfeld der korrespondierende Buchstabe des Geheimtexts auf - im Innern einer Enigma befindet sich also ein Stromkreislauf, der durch das Drücken einer Taste geschlossen wird. Natürlich fliesst der Strom nicht direkt von der Tastatur zum Lampenfeld, sondern über mehrere Umwege.

Von der Tasteneingabe führt der Stromkreislauf erst durch das Steckerbrett. Dieses kannst du dir wie eine alte Telefonzentrale[1] vorstellen: Jeweils zwei Buchstaben werden durch Kabel miteinander verbunden und somit vertauscht. In den meisten Modellen wurden von den möglichen 26 Buchstaben bis zu 10 solche Vertauschpaare gebildet, die restlichen 6 Buchstaben wurden nicht vertauscht.

Danach fliesst der Strom durch die eingesetzten Walzen. Eine Walze ist eine Art Zahnrad mit 26 Zähnen, wobei jeder Zahn einen (elektrischen) Eingang und Ausgang hat. Diese Eingänge werden aber nicht unbedingt direkt weitergeleitet, sondern sind meist mit anderen Ausgängen verdrahtet. Auch hier handelt es sich also um eine Art Substitution, wobei jede Walze eine bestimmte solche Ersetzung vornimmt. Das elektrische Signal führt durch drei solcher Walzen, wird in einer Umkehrwalze 'gespiegelt', und fliesst danach wieder durch die drei Walzen zurück. Darauf folgt wieder das Steckerbrett, wo wiederum allfällige Vertauschungen vorgenommen werden, bevor der Strom im Lampenbrett den entsprechenden Buchstaben des Geheimtexts erleuchtet. [2]

Doch was machte dieses System nun so sicher? Wenn wir wie oben beschrieben Buchstaben miteinander vertauschen (auch wenn wir das mehrmals hintereinander tun), so entspricht dies schlussendlich doch nur einer monoalphabetischen Substitution (analog LATPERM), welche wir mit einer Häufigkeitsanalyse gut knacken könnten. Der Clou liegt daran, dass sich die Walzen bei jeder Eingabe jeweils bewegen, und somit die Verschlüsselung verändern. Für unsere vereinfachte Enigma nehmen wir im Folgenden an, dass die Walzen ähnlich wie ein Kilometerzähler weiterschalten: Nach einer vollständigen Umdrehung der rechten Walze bewegt sich die mittlere, und entsprechend später die linke. Bei der echten Enigma lösten Kerben an den Walzen das Weiterschalten der jeweils nächsten Walze aus. Dadurch entstand unter anderem das sogenannte Double Stepping. Für die Grundidee der Verschlüsselung spielt dieser Unterschied hier aber keine entscheidende Rolle. Die Walze, die ganz rechts eingesetzt wurde, dreht sich jeweils bei jedem Drücken einer Taste, und zwar **bevor** der Strom durch den Kreislauf fliesst. Somit verändert sich auch die entsprechende Verschiebung, und damit die Verschlüsselung. Wenn zweimal hintereinander eine gleiche Taste gedrückt wird, resultiert dies nicht im gleichen Geheimtextbuchstaben, da die rechte Walze die Buchstaben durch die Drehung nun auf eine andere Art vertauscht. Sobald sich die rechte Walze einmal

[1] OK, du hast wahrscheinlich keine Ahnung, wie eine Telefonzentrale funktioniert hat. Jedes Telefon war mit der Telefonzentrale verbunden. Wenn man jemanden anrufen wollte, musste ein Angestellter physisch eine Verbindung zwischen deinem Telefon und demjenigen der Zielperson herstellen.

[2] Diese Geheimtextbuchstaben wurden anschliessend nicht direkt an einen Empfänger übermittelt, sondern notiert und später beispielsweise via Morse-Code an die Zielperson geschickt.

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

vollständig gedreht hat (also nachdem 26 Tasten gedrückt wurden), dreht sich auch die mittlere Walze um eine Position, was wiederum in einer anderen Vertauschung resultiert. Sobald auch die mittlere Walze eine volle Drehung absolviert hat, dreht sich auch die linke Walze.

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Am besten veranschaulichen wir uns die Funktionsweise einer Enigma an einem vereinfachten Exemplar. Wir betrachten eine Enigma, welche lediglich sechs Buchstaben als Eingabe unterstützt und im Steckerbrett zwei Buchstabenpaare vertauschen kann. In der folgenden Abbildung ist die Ausgangslage abgebildet:

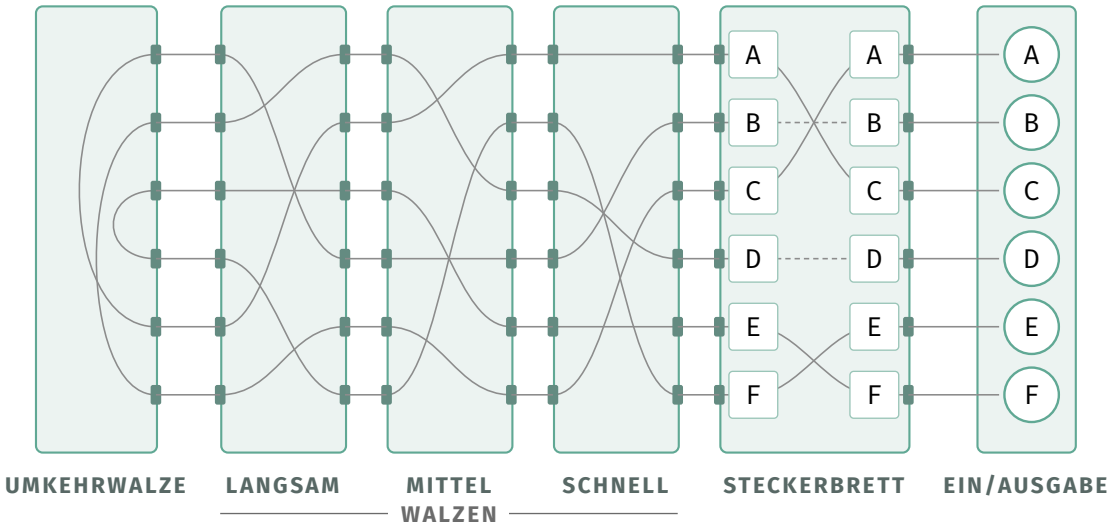

Abbildung 4.2: Vereinfachte Enigma mit sechs Buchstaben

Nun drücken wir die Taste A. Darauf dreht sich die rechte Walze um eine Position. Der Strom fliesst anschliessend durch das Steckerbrett, die Walzen, die Umkehrwalze, zurück durch die Walzen und das Steckerbrett, und lässt schlussendlich den Buchstaben B aufleuchten:

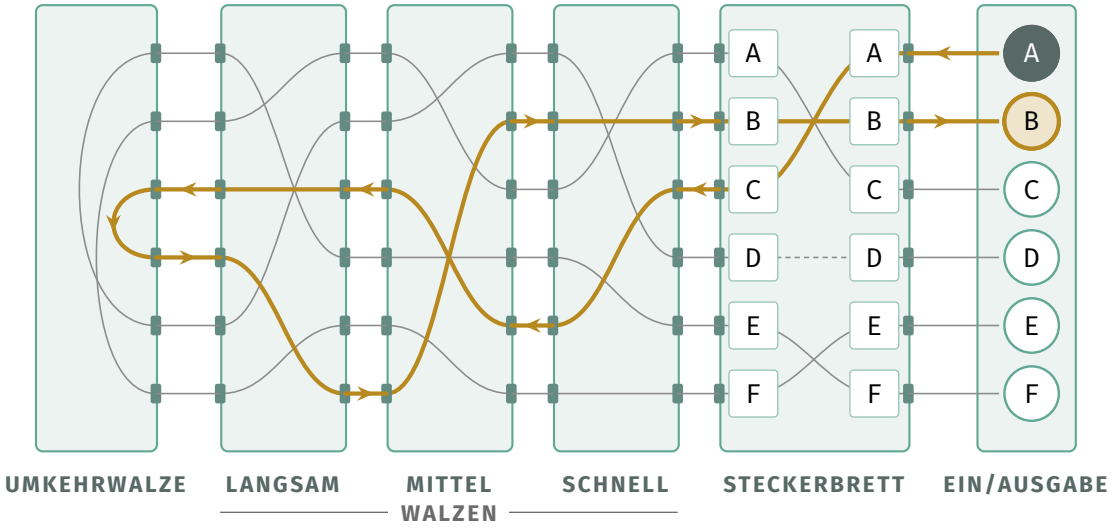

Abbildung 4.3: Ausgangslage nach Drücken der Taste A. Der rechte Rotor dreht sich um eine Position (nach unten) bevor der Strom fliesst. Das A wird also auf ein B abgebildet.

Wenn wir nun ein weiteres Mal die Taste A drücken, so dreht sich der rechte Rotor ein weiteres Mal, die Verschlüsselung ändert sich also erneut. Neu wird A zu D, und nicht wie zuvor zu B. Die Enigma ist also wie Vigenère ein polyalphabetisches Kryptosystem: Ein Buchstabe im Klartext wird nicht immer zwingend auf den gleichen Geheimtextbuchstaben abgebildet.

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

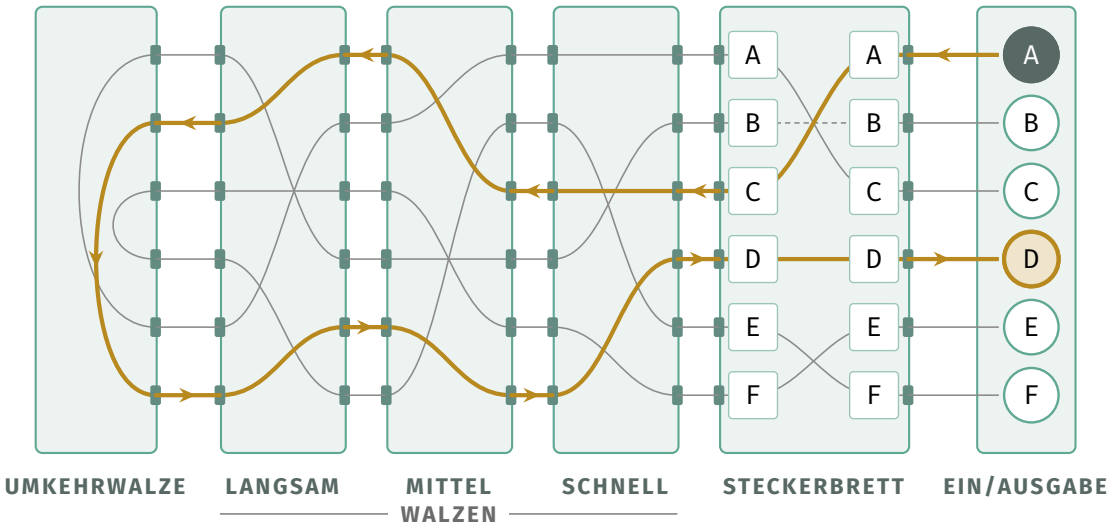

Abbildung 4.4: Durch nochmaliges Drücken der Taste A rotiert die rechte Walze ein weiteres Mal und ändert somit die Verschlüsselung. Aus A wird nun D.

AUFGABE 16: TRIPLE A

Wenn du die Taste A nun ein drittes Mal drückst, welcher Buchstabe würde dann auf dem Lampenbrett aufleuchten?

Entschlüsselung und Schlüsselmenge

Anhand dieses Schemas können wir auch schnell erahnen, wie die Entschlüsselung eines mit der Enigma verschlüsselten Geheimtexts funktioniert. Bei einem Tastendruck bildet sich ein geschlossener Stromkreislauf zwischen zwei Buchstaben. Je nachdem, von welcher Seite der Strom nun fliesst, wird der eine oder der andere Buchstabe dieses Paares auf dem Lampenbrett aufleuchten. Um einen Geheimtext zu entschlüsseln, tippt man diesen einfach nochmals auf einer Enigma ein. Wichtig dabei ist aber, dass sich die Enigma zu Beginn der Verschlüsselung bzw. Entschlüsselung im exakt gleichen Zustand befindet. Ein solcher ist definiert durch die eingesetzten Walzen, deren initiale Rotationsposition sowie die gebildeten Paare auf dem Steckerbrett.

Jeder dieser Zustände ist ein Schlüssel im Kryptosystem, die Anzahl Zustände entspricht also der Grösse der Schlüsselmenge. Wir berechnen die Schlüsselmenge für unser vereinfachtes Modell. Details wie die Ringstellung lassen wir dabei vorerst ausser Acht. Wie viele Schlüssel sind das denn nun? Wie wir gleich sehen werden: sehr, sehr viele.

Betrachten wir zuerst die Anzahl Möglichkeiten, die sich für das Einlegen der Walzen ergeben. Hierbei können aus fünf verschiedenen Walzen deren drei ausgewählt und in die Enigma eingesetzt werden. Für die linke (langsame) Walze hat man also 5 mögliche Walzen zur Auswahl, für die mittlere Walze deren 4 und für die schnelle rechte Walze noch 3 Optionen. Total sind das also $5 \cdot 4 \cdot 3 = 60$ Möglichkeiten für die **Walzenlage**.

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Jede dieser Walzen kann nun unabhängig von den anderen in eine von 26 Positionen rotiert werden. Dies ergibt $26^3 = 17\,576$ Möglichkeiten für die so genannte **Grundstellung** der eingesetzten Walzen.

VERTIEFUNG: WOHER KOMMEN DIE 150 738 274 937 250 MÖGLICHKEITEN?

Die mit Abstand meisten Möglichkeiten ergeben sich beim **Steckerbrett**, wo wir aus den total 26 Buchstaben 10 Paare bilden können. Wenn du wissen möchtest, wie diese Zahl zustande kommt, können wir sie mit etwas Kombinatorik herleiten.
Wir betrachten zuerst das erste Buchstabenpaar: Für den ersten Stecker haben wir 26 Möglichkeiten zur Auswahl, für den zweiten noch deren 25. Da es aber keine Rolle spielt, in welcher Reihenfolge wir dies tun, also ob wir beispielsweise G und E oder E und G miteinander verbinden, müssen wir diese Anzahl Möglichkeiten noch halbieren, und erhalten somit $\frac{26 \cdot 25}{2}$ mögliche erste Paare. Analog ergeben sich für das zweite Paar $\frac{24 \cdot 23}{2}$ Möglichkeiten, für das dritte deren $\frac{22 \cdot 21}{2}$, usw. Wenn wir das für alle zehn Paare weiterführen, erhalten wir

$$\prod_{i=1}^{10} \frac{(26 - 2 \cdot i + 2) \cdot (26 - 2 \cdot i + 1)}{2}$$

Möglichkeiten. Da es schlussendlich aber egal ist, in welcher Reihenfolge wir die jeweiligen Paare gebildet haben, müssen wir noch durch die Anzahl möglicher Permutationen der Paare teilen. Bei 10 Paaren sind dies 10! mögliche Anordnungen. Die Anzahl Möglichkeiten, auf dem Steckerbrett zehn Buchstabenpaare zu bilden, ergibt sich folglich als

$$\frac{1}{10!} \prod_{i=1}^{10} \frac{(26 - 2 \cdot i + 2) \cdot (26 - 2 \cdot i + 1)}{2} = \frac{26!}{2^{10} \cdot 10! \cdot 6!} = 150\,738\,274\,937\,250$$

Die **Anzahl möglicher Schlüssel** ergibt sich nun, wenn wir diese drei Komponenten miteinander kombinieren:

$$60 \cdot 17\,576 \cdot 150\,738\,274\,937\,250 = \mathbf{158\,962\,555\,217\,826\,360\,000}$$

verschiedene Einstellungen sind also möglich. Eine riesige Zahl! Es ist schwierig, sich die Grösse dieser Zahl vorzustellen. Der folgende Vergleich soll dies verdeutlichen.

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

6 Schlüsselaustausch

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

Ein Aspekt, den wir bis anhin ausgeblendet haben, ist die Frage, wie zwei Parteien überhaupt zu einem gemeinsamen Schlüssel kommen, damit sie verschlüsselte Nachrichten austauschen können.

Einen geheimen Schlüssel einfach so ungesichert auszutauschen, ist definitiv keine gute Idee. Jede und jeder, der diese Schlüsselnachricht abfangen kann, ist danach in der Lage, die gesamte verschlüsselte Kommunikation mitzulesen. Den Schlüssel selber verschlüsseln können wir auch nicht[1], da uns dazu ja eben ein gemeinsamer Schlüssel fehlt. Eine abhörsichere Telefon- oder Internetleitung hat ja auch nicht jeder, und wenn man sich für jede geheime Kommunikation erst an einem geheimen Ort treffen muss, um einen Schlüssel auszutauschen, so ist das alles andere als praktikabel!

Am einfachsten wäre es natürlich, wenn wir eine Art **Schlüsselaustausch** über einen unsicheren Kanal[2] hinbekommen würden. Aber ist das überhaupt möglich? Die erstaunliche Antwort ist: Ja, das ist möglich!

Zwei Schlösser, eine Truhe

Wir schauen uns zuerst ein Beispiel aus der analogen Welt an. Alice und Bob haben je ein (unterschiedliches) Vorhängeschloss zur Hand. Wir nehmen hier an, dass es sich um Zahlenschlösser handelt. Die sind zwar mit genügend Zeitaufwand einfach knackbar, aber so vermeiden wir, dass der Begriff Schlüssel hier verwechselt wird: Der Schlüssel ist das, was wir austauschen möchten und nicht das, was die jeweiligen Schlösser öffnet. Mit diesen Schlössern können sie eine Truhe so verschliessen, dass niemand den Inhalt begutachten

[1] Bis jetzt nicht, mit dem später vorgestellten Konzept der Public-Key-Kryptologie ist das theoretisch möglich.

[2] Mit unsicher meinen wir hier einen nicht abhörsicheren Kanal, also einen, bei dem theoretisch alle mithören können.

kann. Die Truhe selbst können sie via Post untereinander austauschen, aber wenn die Truhe nicht verschlossen ist, kann jeder Postbote einen Blick hinein werfen!

DENKANSTOSS: ZWEI SCHLÖSSER, EINE TRUHE

Alice möchte Bob einen geheimen Schlüssel schicken. Die Truhe darf beliebig oft zwischen ihnen hin- und hergeschickt werden. Alice und Bob besitzen je ein eigenes Vorhängeschloss, können das Schloss der anderen Person aber nicht öffnen.
Findest du eine Möglichkeit, wie der Schlüssel bei Bob ankommt, ohne dass die Truhe jemals unverschlossen durch das Postnetz geschickt wird?

Wenn sie diese Schlösser geschickt einsetzen, können sie damit wie folgt einen geheimen Schlüssel für zukünftige geheime Nachrichten austauschen: Alice legt den Schlüssel, den sie später zum Nachrichtenaustausch verwenden will, in die Truhe und verschliesst diese mit ihrem Schloss. Die Truhe schickt sie nun an Bob, der nun sein Schloss ebenfalls um die Truhe legt: Die Truhe ist also nun doppelt verschlossen! Bob schickt die Truhe zurück an Alice, diese entfernt ihr Schloss und schickt die Truhe zurück. Auch jetzt kann kein neugieriger Postbote den Inhalt einsehen, die Truhe ist immer noch mit Bobs Schloss versiegelt! Bob kann dieses aber nach Empfang der Truhe entfernen und kommt so zum geheimen Schlüssel. Ziel erreicht! Dieses Vorgehen ist in Abbildung 6.1 dargestellt:

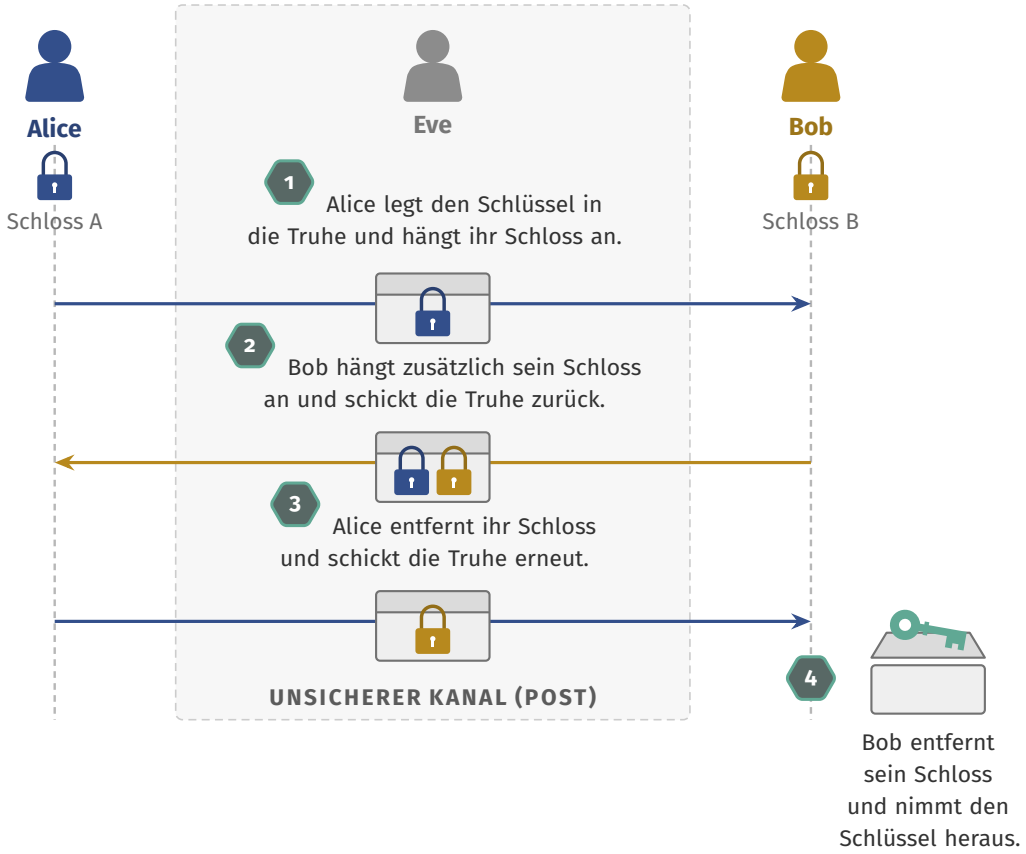

Abbildung 6.1: Analoger Schlüsselaustausch mit einer Truhe und Vorhängeschlössern

Wie du siehst: Zu keiner Zeit geht die Truhe unverschlüsselt über den unsicheren Kanal, hier

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

das Postnetz. Ein neugieriger Postbote, oder besser eine neugierige Postbotin, hat so niemals Einblick in die Truhe und keine Ahnung, was darin steckt. Die mithörende Dame zwischen Alice und Bob nennt man nämlich meist Eve.

Für den Moment geben wir Eve nur Ohren, aber keine Hände: Sie darf jede Nachricht mitlesen, die zwischen Alice und Bob ausgetauscht wird. Verändern, löschen oder durch eigene Nachrichten ersetzen darf sie noch nichts. Später geben wir ihr ein bisschen mehr Macht und schauen, was dann passiert.

Können wir diese Idee auch für den digitalen Schlüsselaustausch verwenden? Wir wollen es versuchen! Statt Vorhängeschlösser zur Verschlüsselung wollen wir das perfekt sichere One-Time-Pad-Verfahren in seiner binären Form verwenden. Alice wählt einen binären Schlüssel, also eine Folge von Bits, die sie zusammen mit Bob später für ein digitales Kryptosystem verwenden kann. Wir nennen diesen zukünftigen Schlüssel hier Schlüsselnachricht (m), um Verwechslungen vorzubeugen. Verschlüsselte Nachrichten beginnen wir mit c, aus dem englischen ciphertext.

Alice wählt zuerst einen passenden (binären) OTP-Schlüssel (k_a), den sie für sich behält. Damit verschlüsselt sie die geheime Schlüsselnachricht und schickt das Ergebnis $c_a = m \oplus k_a$ an Bob. Dieser verschlüsselt die Nachricht nochmals mit seinem eigenen OTP-Schlüssel (k_b) und schickt die nun doppelt verschlüsselte Schlüsselnachricht $c_{ab} = c_a \oplus k_b$ zurück. Alice entschlüsselt nun mit ihrem k_a und schickt das Ergebnis $c_b = c_{ab} \oplus k_a$ an Bob. Dieser entschlüsselt nun mit seinem k_b und erhält $m' = c_b \oplus k_b$ als Resultat. Abbildung 6.2 stellt dieses Schlüsselaustauschprotokoll graphisch dar:

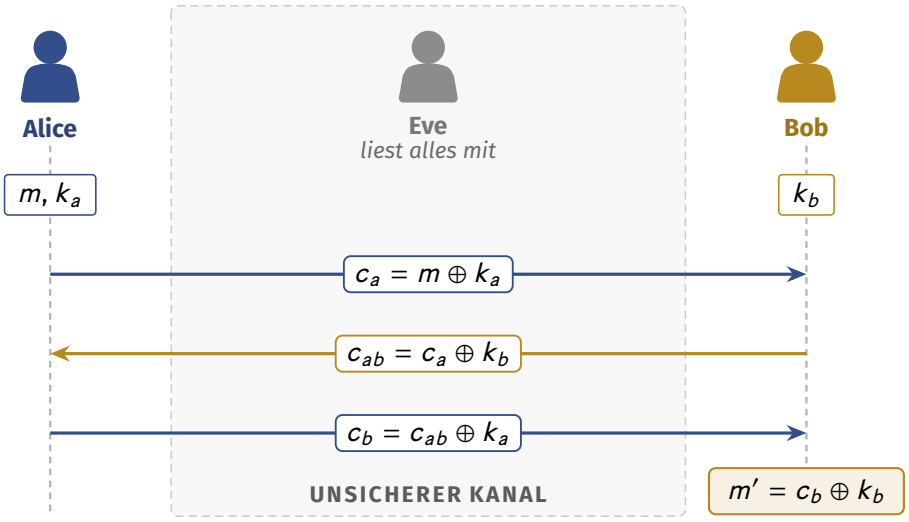

Abbildung 6.2: Digitaler Schlüsselaustausch mit dem OTP-Verfahren

Um herauszufinden, ob dieses Verfahren tatsächlich funktioniert, müssen wir zwei Fragen beantworten:

1. Erhält Bob am Schluss wirklich die ursprüngliche Schlüsselnachricht m?
2. Kann Eve aus den drei mitgelesenen Nachrichten etwas über m herausfinden?

Die erste Frage überlasse ich für einen Moment dir.

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ

DENKANSTOSS: ERHÄLT BOB WIRKLICH m?

Setze für c_b, c_{ab} und c_a nacheinander ihre Definitionen ein. Kannst du zeigen, dass am Schluss $m' = m$ übrig bleibt?

Zuerst wollen wir zeigen, dass $m' = m$ gilt. Dazu ersetzen wir zuerst nach und nach einige Terme mit ihrer Definition:

$$\begin{aligned} m' &= c_b \oplus k_b \\ &= (\underbrace{c_{ab} \oplus k_a}_{c_b}) \oplus k_b \\ &= ((\underbrace{c_a \oplus k_b}_{c_{ab}}) \oplus k_a) \oplus k_b \\ &= (((\underbrace{m \oplus k_a}_{c_a}) \oplus k_b) \oplus k_a) \oplus k_b \end{aligned}$$

Da der XOR-Operator $\oplus$ assoziativ ist, können wir die Klammern beliebig setzen, oder auch ganz weglassen. Die Kommutativität erlaubt es uns zudem, die einzelnen Terme so zu ordnen, wie es uns gefällt. Damit können wir nun weiter rechnen:

$$\begin{aligned} m' &= (((m \oplus k_a) \oplus k_b) \oplus k_a) \oplus k_b \\ &= m \oplus k_a \oplus k_b \oplus k_a \oplus k_b \\ &= m \oplus (\underbrace{k_a \oplus k_a}_{\mathbf{0}}) \oplus (\underbrace{k_b \oplus k_b}_{\mathbf{0}}) \\ &= m \oplus \mathbf{0} \oplus \mathbf{0} \\ &= m \end{aligned}$$

Et voilà: Die Nachricht m', welche Bob am Schluss erhält, entspricht genau jener, die Alice zu Beginn verschickt hat (m). Oder anders ausgedrückt: Alice kann damit tatsächlich einen Schlüssel an Bob schicken, und dieser erhält schlussendlich genau diesen Schlüssel. Die erste Bedingung an unseren Schlüsselaustausch ist also erfüllt.

Perfekt, Bob bekommt also tatsächlich den richtigen Schlüssel. Damit ist aber erst die Hälfte der Arbeit erledigt. Eve kennt c_a, c_{ab} und c_b.

Eve könnte zum Beispiel also mal alle dieser Nachrichten zusammenzählen, bzw. XOR-en. Mal schauen, was dabei herauskommt! Zuerst ersetzen wir wiederum einzelne Terme mit ihrer Definition und ordnen diese danach etwas um:

$$\begin{aligned} c_a \oplus c_{ab} \oplus c_b &= (\underbrace{m \oplus k_a}_{c_a}) \oplus (\underbrace{m \oplus k_a \oplus k_b}_{c_{ab}}) \oplus (\underbrace{m \oplus k_a \oplus k_b \oplus k_a}_{c_b}) \\ &= m \oplus (\underbrace{m \oplus m}_{\mathbf{0}}) \oplus (\underbrace{k_a \oplus k_a}_{\mathbf{0}}) \oplus (\underbrace{k_a \oplus k_a}_{\mathbf{0}}) \oplus (\underbrace{k_b \oplus k_b}_{\mathbf{0}}) \\ &= m \oplus \mathbf{0} \oplus \mathbf{0} \oplus \mathbf{0} \oplus \mathbf{0} \\ &= m \end{aligned}$$

LESEPROBE · INF.SCHOOL · EINSATZ IM UNTERRICHT NUR MIT LIZENZ