wienerwaldgymnasium

Kryptografie-Entwicklung und Anwendung der mathematischen Techniken

Vorwissenschaftliche Arbeit

von

Julia Klarissa Grün
Klasse 9O

BetreuerIn: Mag. Clarissa Friedrichkeit-Miko

Abgabetermin: 16. Februar 2024

BG/BRG Tullnerbach
Norbertinumstraße 7, 3013 Tullnerbach

Abstract

Kryptografie hat eine sehr große Bedeutung in der heutigen technologischen Welt. Das Ziel dieser Arbeit ist es, die unterschiedlichen asymmetrischen Methoden der Kryptografie zu vergleichen, deren Vor- und Nachteile zu beleuchten und die historische Entwicklung der Kryptografie zu beschreiben. Die untersuchten asymmetrischen Kryptografieverfahren sind der Diffie-Hellman-Merkle-Schlüsseltausch, das RSA-Kryptosystem, die Elliptische-Kurven-Kryptografie und das Softwareprogramm Pretty Good Privacy. Um sich mit den Methoden der Kryptografie näher befassen zu können, werden zuerst die notwendigen mathematischen Techniken vorgestellt. Die vorliegende Arbeit beantwortet die Fragen, wo und wie die Kryptografie im digitalen Bereich heute angewandt wird, und widmet sich im Speziellen dann auch den Sicherheitsaspekten von ausgewählten kryptografischen Verfahren. Mit einem Ausblick in welche Richtung sich die Kryptografie weiter entwickeln wird, schließt die Arbeit ab. Dabei wird insbesondere auf quantentechnologische Aspekte eingegangen. Unterstützt werden die Betrachtungen durch einige konkrete Beispiele, zum Ver- und Entschlüsseln von Informationen. Als eines der fundamentalen Ergebnisse der Arbeit hat sich herausgestellt, dass es bei der Auswahl des passenden Kryptografieverfahren auf die geplante Anwendung ankommt. Denn mit zunehmender Sicherheit, bis zur perfekten Sicherheit, nehmen die Nachteile, wie zum Beispiel die Berechnungsdauer und andere Umsetzungsschwierigkeiten zu.

Inhaltsverzeichnis

1. Einleitung

Viele Menschen sind der Ansicht, Mathematik ist nur zum Lösen von einfachen Rechenaufgaben zu gebrauchen und sonst reine Spielerei. Jedoch brauchen wir, auch wenn wir es nicht bewusst wahrnehmen, Mathematik täglich. Sei es beim Senden von Nachrichten, dem Surfen im Internet oder dem Überweisen von Geld - das sind alles Tätigkeiten, welche einer Verschlüsselung, beruhend auf mathematischen Grundlagen, bedürfen. Aufgrund dieser aktuellen großen Relevanz dieses Themas, befasst sich die vorliegende Arbeit mit den mathematischen Techniken, welche vor allem bei asymmetrischen Kryptografieverfahren verwendet werden.

Zuerst wird die Frage beantwortet, wo und wie Kryptografie damals und heute angewandt wird, um anschließend der Frage nachzugehen, wie sich die Kryptografie im Laufe der Zeit entwickelt hat. Dabei wird auch auf das Beispiel One-Time-Pad genauer eingegangen und anhand eines eigenen Beispiels die Ver- und Entschlüsselung vorgeführt und die Sicherheit dieses Verschlüsselungsverfahren erklärt. In weiterer Folge wird die asymmetrische Kryptografie untersucht, wobei vor allem konkrete Verschlüsselungsmethoden verglichen und die jeweiligen Vor- und Nachteile inklusive des Sicherheitsaspekts beleuchtet werden. Die hier untersuchten asymmetrischen Kryptografieverfahren sind der Diffie-Hellman-Merkle-Schlüsseltausch, das RSA-Kryptosystem, die Elliptische-Kurven-Kryptografie und das Softwareprogramm Pretty Good Privacy. Am Schluss werden noch mögliche Antworten nach der weiteren Entwicklung der Kryptografie gegeben, wobei in diesem Kapitel nicht nur auf die Quantenkryptografie eingegangen, sondern auch die Postquantenkryptografie einbezogen wird.

Beantwortet werden die gestellte Fragen vordergründig mit Literatur, wobei sich der Bogen von Erklärungen für Jugendliche bis zu wissenschaftlichen Schriften spannt. Bei der anspruchsvolleren Literatur reicht die Herangehensweise an das Thema von einer rein mathematischen Sicht bis zu informationstechnologischen Umsetzungsmöglichkeiten. Zusätzlich zur Literatur sollen selbst überlegte Beispiele für jeweils eine Ver- und Entschlüsselung tiefere Einblicke, vor allem bezüglich der Sicherheit, ermöglichen. Diese Praxisanwendungen zu asymmetrischen Kryptografieverfahren wurden für den Diffie-Hellmann-Merkle-Schlüsseltausch und das RSA-Kryptosystem erstellt.

2. Mathematische Grundlagen

2.1 Modulo

Die Rechenoperation Modulo berechnet den Rest einer Division zweier Zahlen. Diese Rechenoperation wird in Rechnungen meist mit mod abgekürzt. Ein Beispiel wäre $13 \bmod 3 = 1$. An diesem Beispiel ist zu erkennen, dass es nicht möglich ist, vom Rest (1) und dem Divisor (3) auf den Dividenden (13) zu schließen. (Beutelspacher, Kryptologie, 2015)

2.2 Primzahlen

Primzahlen gehören den natürlichen Zahlen an und haben nur zwei Teiler, eins und sich selbst. Alle Zahlen größer 1, welche selbst keine Primzahl sind, können als Produkt von mindestens 2 Primzahlen aufgeschrieben werden. Betragsmäßig hohe Primzahlen können, auch nicht mit Computern, effizient gefunden werden. (Beutelspacher, Kryptologie, 2015)

3. Anwendungsbereiche der Kryptografie

Früher wurde die Kryptografie vor allem für militärische und politische Zwecke verwendet. Heute ist Kryptografie aus dem alltäglichen Leben von jedem Menschen nicht mehr weg zu denken. Beispiele für Kryptografie im Alltag sind das Internet, das WLAN, der Autoschlüssel, die Bankomatkarten, Emails oder Telefonate. Darüber hinaus wird Kryptografie heute auch noch vom Militär, der Politik und privaten Unternehmen angewendet, um zum Beispiel Firmengeheimnisse zu wahren oder einfach nur aus Datenschutzgründen.

4. Geschichte der Symmetrischen Verschlüsselungsverfahren

4.1 Altertum, Mittelalter und Neuzeit

Der früheste bekannte Einsatz von Kryptografie fand im alten Ägypten statt. Dort wurden für bestimmte Texte ein anderer Satz Hieroglyphen verwendet. Auch die Griechen und die Römer setzten auf Verschlüsselung. Während die Spartaner mittels eines Holzstabes Transpositionen durchführten, verwendete vor allem Julius Cäsar, die nach ihm benannte Cäsar-Chiffre, welche auf Substitution beruht (Singh, 2000), (Beutelspacher, Kryptologie, 2015), (Kippenhahn, 2012), (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010). Im Mittelalter wurde hauptsächlich in der arabischen Welt verschlüsselt. In dieser Zeit wurde auch das erste Buch zu diesem Thema von einem islamischen Theologen und Philosophen geschrieben. In der Neuzeit hatte die Kryptografie wieder einen Aufschwung. Zum Beispiel entwickelte Blaise de Vigenère die nach ihm benannte Vigenère-Chiffre. Dabei wird ein Text, Buchstabe für Buchstabe mittels eines Passworts substituiert (Beutelspacher, Geheimsprachen, 2012), (Ertel & Löhmann, 2020).

4.2 1. Weltkrieg bis nach dem 2. Weltkrieg

Während des ersten Weltkrieges wurden (eher einfache) Verschlüsselungsverfahren eingesetzt, welche händisch durchgeführt wurden. Das bekannteste von ihnen heißt ADFGX und wurde vom deutschen Militär verwendet. Die Verschlüsselung besteht aus zwei hintereinander ablaufenden Schritten, wobei ersterer die Nachricht substituiert und in zweiterem die Nachricht durch Transposition verschlüsselt wird (Singh, 2000). Ziel war es auf der einen Seite, die Nachrichten immer sicherer zu verschlüsseln und auf der anderen Seite, gab es die Bemühungen immer mehr verschlüsselte Nachrichten der Gegner zu entschlüsseln, wobei Letzteres oftmals mehr Erfolg hatte. Die gemachten Erfahrungen, während des ersten Weltkrieges, führten noch gegen Ende des Krieges zu den ersten Erfindungen von Verschlüsselungsmaschinen. Diese waren um einiges sicherer als die händischen Verschlüsselungsmethoden. Trotzdem wurden die manuellen Methoden von den Maschinen noch nicht verdrängt, da letztere sehr teuer waren und somit anfangs nur für sehr wichtige Nachrichten verwendet wurden.

Im zweiten Weltkrieg wurden von allen beteiligten Parteien verschiedene Verschlüsselungsverfahren, welche vor allem durch Maschinen durchgeführt wurden, angewandt. Hier ist auf deutscher Seite die Enigma erwähnenswert. Die Maschine galt bei den Deutschen als nicht entschlüsselbar, doch gelang dies den Gegnern. Die Enigma wurde zuerst von den Polen entschlüsselt, welche ihre Erkenntnisse mit ihren britischen und französischen Verbündeten teilten. An die Erfolge der Polen anknüpfend, entwickelten die Briten sogar eine eigene Maschine für die Entschlüsselung der Enigma und konnten daraufhin ab Jänner 1940 fast den ganzen Funkverkehr mitlesen (Singh, 2000), (Beutelspacher, Geheimsprachen, 2012). Mit den Jahren

verbesserten die Deutschen die Enigma immer wieder. Daraufhin mussten die britischen Kryptoanalytiker weitere neue Maschinen erfinden und bauen. Gegen Ende des Krieges entwickelten die Briten 1943 hierfür einen Art ersten programmierbaren Computer namens Colossus (Ertel & Löhmann, 2020). Auf amerikanischer Seite wurden bei der Verschlüsselung von Nachrichten, neben Maschinen, auch auf die Sprache Navajo gesetzt, welche die Muttersprache einiger nordamerikanischer Ureinwohnerstämme ist (Singh, 2000). Auf der Entschlüsselungsseite hatte die USA ebenso große Erfolge, zum Beispiel bei der Entzifferung japanischer Funksprüche (Singh, 2000).

Während des Kalten Krieges setzte die Sowjetunion bei der Verschlüsselung von Nachrichten vor allem auf das perfekte Sicherheit bietende Verschlüsselungsverfahren namens One-Time-Pad. Wobei perfekte Sicherheit bedeutet, dass das Verfahren, wenn man bestimmte Regeln beachtet, nicht zu brechen ist. Doch machten sie dabei einige Fehler, welche die Sicherheit beeinträchtigten und es der Gegenseite möglich machten, Nachrichten zu entschlüsseln: manche geheime Schlüssel wurden öfters wiederverwendet. Im folgenden Kapitel wird dieses Verschlüsselungsverfahren etwas genauer beschrieben.

4.3 One-Time-Pad

Dieses symmetrische Verschlüsselungsverfahren wurde 1917 von Gilbert S. Vernam erfunden. Das Verfahren beruht auf Substitution, wobei der Schlüssel zufällig generiert wird und jeweils nur einmal verwendet werden darf. Werden diese Regeln befolgt, so gilt die Verschlüsselung als perfekt, das heißt, es ist nicht möglich von dem verschlüsselten Text auf die entschlüsselte Nachricht schließen. Früher wurden die sehr langen Schlüssel oft von Menschen erzeugt und dann auf Abreißblöcke gedruckt – daher auch der Name One-Time-Pad, oder kurz OTP. Leider enthielten die von Menschen erzeugten Schlüssel bestimmte Muster und waren nicht komplett zufällig. Außerdem wurden manchmal Schlüssel wiederverwendet. Mithilfe dieser Fehler war es Angreifern immer wieder möglich, die Verschlüsselung zu brechen (Beutelspacher, Kryptologie, 2015).

4.3.1 Theoretische Durchführung

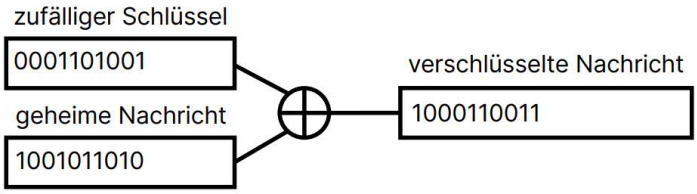

Abbildung 1: Funktionsweise der XOR-Verknüpfung für das OTP

Wie bei jeder symmetrischen Verschlüsselung muss zuerst ein Schlüssel erzeugt werden. Bei dem One-Time-Pad generiert der Sender zuerst einen großen geheimen Schlüssel aus Zahlen, Groß-

und Kleinbuchstaben mittels eines Zufallsgenerators. Anschließend teilt der Absender diesen geheimen Schlüssel dem Empfänger auf einem sicheren Weg mit. Nun übersetzt der Sender die geheime Botschaft zum Beispiel mittels ASCII-Code in eine binäre Zahlenfolge. Dies macht er auch mit dem zuvor generierten Schlüssel. Als Nächstes addiert der Absender die beiden binären Zahlenfolgen bitweise, wobei $1 + 1 = 0$ gilt (siehe Abbildung 1). Abschließend kann der Sender die erhaltene Zahlenfolge mithilfe des ASCII-Code wieder in Buchstaben und Zahlen umwandeln oder die binäre Zahlenfolge versenden. Anschließend sollten alle Aufzeichnungen über den verwendeten Schlüssel unwiderruflich vernichtet werden.

Der Empfänger erhält also entweder die binäre Zahlenfolge oder den in Buchstaben und Zahlen umgewandelten Text. Letzteres wandelt der Empfänger vor dem Entschlüsseln zum Beispiel mittels ASCII-Code in eine binäre Zahlenfolge um. Im nächsten Schritt muss der Empfänger, den vom Absender geheim erhaltenen Schlüssel auf die gleiche Art bearbeiten. In Folge subtrahiert er bitweise die binäre Zahlenfolge des Schlüssels von jener der verschlüsselten binären Nachricht, wobei 0-1=1 gilt. Die daraus resultierende binäre Zahlenreihe übersetzt der Empfänger, zum Beispiel mit Hilfe des ASCII-Code, wieder in Buchstaben und Zahlen.

Generell gilt, dass Sender und Empfänger im Vorhinein einige Dinge absprechen müssen. Darunter fällt die Methode wie aus Buchstaben und Zahlen eine binäre Zahlenreihe zu erzeugen ist (Ertel & Löhmann, 2020), (Schmeh, 2016).

4.3.2 Praktisches eigenes Beispiel

Der Absender einer geheimen Nachricht generiert zufällig einen geheimen Schlüssel. Dann überlegt sich der Sender der geheimen Nachricht zuerst die zu verschlüsselnde Mitteilung (siehe Abbildung 2 Tabellenzeile „geheime Mitteilung“). Anschließend wird dieser mittels ASCII-Tabelle in binäre Zahlen übersetzt (siehe Abbildung 2 Tabellenzeile „geheimen Nachricht“). Im Weiteren verknüpft der Absender die geheime binäre Nachricht mit dem geheimen binären Schlüssel gemäß den nachfolgenden Regeln und rechnet das binäre Ergebnis in Dezimalzahlen um (siehe Abbildung 2 Tabellenzeilen „verschlüsselte Nachricht“ und „verschlüsselte Dezimalzahlen“).

XOR-bitweise Verknüpfungsregeln:

$1 \oplus 1 = 0 \qquad 1 \oplus 0 = 1 \qquad 0 \oplus 1 = 1 \qquad 0 \oplus 0 = 0$

Zufälliger Schlüssel	1111110	1001010	0001100	0111001	0101000	0011101	1011010	0000100	1110000
Geheime Mitteilung	g	e	h	e	i	m	n	i	S
Geheime Nachricht	1100111	1100101	1101000	1100101	1101001	1101101	1101110	1101001	1110011
verschlüsselte Nachricht	0011001	0101111	1100100	1011100	1000001	1110000	0110100	1101101	0000011
Verschlüsselte Dezimalzahlen	025	047	100	092	065	112	052	109	003

Abbildung 2: Berechnungen des Absenders beim One-Time-Pad

Abschließend übermittelt der Sender dem Empfänger die verschlüsselten Dezimalzahlen.

Der Empfänger erhält vom Absender den geheimen Schlüssel (siehe Abbildung 3 Tabellenzeile „geheimer Schlüssel“) und die verschlüsselten Dezimalzahlen (siehe Abbildung 3 Tabellenzeile „verschlüsselte Dezimalzahlen“). Diese Zahlen rechnet der Empfänger zuerst in binäre Zahlen (siehe Abbildung 3 Tabellenzeile „verschlüsselte Nachricht“) um. Anschließend verknüpft er den geheimen Schlüssel und die verschlüsselte Nachricht nach den Regeln der XOR-Verknüpfung. Abschließend muss der Empfänger die entschlüsselten binären Zahlen (siehe Abbildung 3 Tabellenzeile „entschlüsselte Nachricht“) mittels ASCII-Tabelle in Buchstaben (siehe Abbildung 3 Tabellenzeile „entschlüsselter Text“) umwandeln.

Geheimer Schlüssel	1111110	1001010	0001100	0111001	0101000	0011101	1011010	0000100	1110000
Verschlüsselte Dezimalzahlen	025	047	100	092	065	112	052	109	003
Verschlüsselte Nachricht	0011001	0101111	1100100	1011100	1000001	1110000	0110100	1101101	0000011
Entschlüsselte Nachricht	1100111	1100101	1101000	1100101	1101001	1101101	1101110	1101001	1110011
Entschlüsselter Text	g	e	h	e	i	m	n	i	S

Abbildung 3: Berechnungen des Empfängers beim One-Time-Pad

4.3.3 Anwendung

Da die binäre Zahlenfolge des Schlüssels genau so lang sein muss, wie jene der Nachricht, ist der Schlüssel meist sehr lang. Somit ist es recht schwer den Schlüssel dem Empfänger der Nachricht sicher mitzuteilen. Aufgrund dieser Schwierigkeit wurde dieses Verschlüsselungsverfahren in der Vergangenheit nur für wirklich wichtige Nachrichten verwendet. Zum Beispiel wurde dieses Kryptografieverfahren während des 2. Weltkrieg für die Kommunikation zwischen den englischen Codeknackern und dem damaligen Premierminister verwendet, um selbst entzifferte Nachrichten sicher zu übermitteln (Beutelspacher, Geheimsprachen, 2012).

4.4 Verschlüsselung mit Computer ab 1970

Im Laufe der Zeit wurde aus der geheim betriebenen Wissenschaft eine allgemein zugängliche Forschungsdisziplin. Mit den Bemühungen eine behördenübergreifende Verschlüsselung in der USA zu etablieren, wurde das DES (Data Encryption Standard) Verfahren entwickelt. Dieses besteht, in dieser und weiterentwickelten Formen, bis heute und wird beispielsweise noch von Banken verwendet (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010). Aufgrund der immer besser werdenden Computer und der damit schnelleren Entzifferung von symmetrischen Verschlüsselungsverfahren, wurde im Jänner 1997 angefangen, einen Nachfolger für das DES zu finden. In diesem Prozess wurden erstmals Vorschläge aus aller Welt entgegengenommen. Der Sieger der ursprünglich 15 Vorschläge, hieß Rijndael und war seinen Konkurrenten vor allem in der Geschwindigkeit überlegen. Somit wurde Rijndael als Nachfolger des DES-Algorithmus als AES (Advanced Encryption Standard) berühmt (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010), (Ertel & Löhmann, 2020).

5. Elektronische / Asymmetrische Verfahren

5.1 Diffie-Hellman-Merkle-Schlüsseltausch

Die Lösung für das Problem des geheimen Schlüsselaustauschs entdeckte zuerst Malcolm Williamson mit Vorarbeit seiner Kollegen beim britischen Geheimdienst im Jahre 1974. Wobei das Problem des geheimen Schlüsselaustausches sehr gut mit folgendem Zitat beschrieben werden kann.

> Das ganze Problem der Schlüsselverteilung ist eine klassische Paradoxie. Wenn ein Mensch einem anderen eine geheime Nachricht […] übermitteln will, muss er sie verschlüsseln. Dazu braucht er einen Schlüssel, der selbst wiederum ein Geheimnis ist, und so ergibt sich das Problem, diesen geheimen Schlüssel dem Empfänger zu übermitteln, damit die geheime Botschaft gesendet werden kann. Kurz, wenn zwei Menschen sich ein Geheimnis (eine verschlüsselte Botschaft) mitteilen wollen, müssen sie sich zuvor bereits ein Geheimnis (den Schlüssel) mitgeteilt haben (Singh, 2000, S. 311).

Jedoch wurde die Erfindung unter Verschluss gehalten, da sie für den britischen Geheimdienst zu wichtig war. Dies hatte zur Folge, dass die Lösung 1976 nochmals von amerikanischen Wissenschaftlern gefunden wurde (Singh, 2000). Nachdem die erste Entdeckung nicht öffentlich bekannt war, wurde das Verfahren Diffie-Hellman-Merkle-Schlüsselaustausch oder auch kurz DH, nach den amerikanischen Wissenschaftlern Whitfield Diffie, Martin Hellman und Ralph Merkle benannt. Das Verfahren ermöglicht es zwei bisher fremden Personen miteinander über öffentliche Wege einen geheimen gemeinsamen Schlüssel zu erzeugen (Singh, 2000).

5.1.1 Veranschaulichung

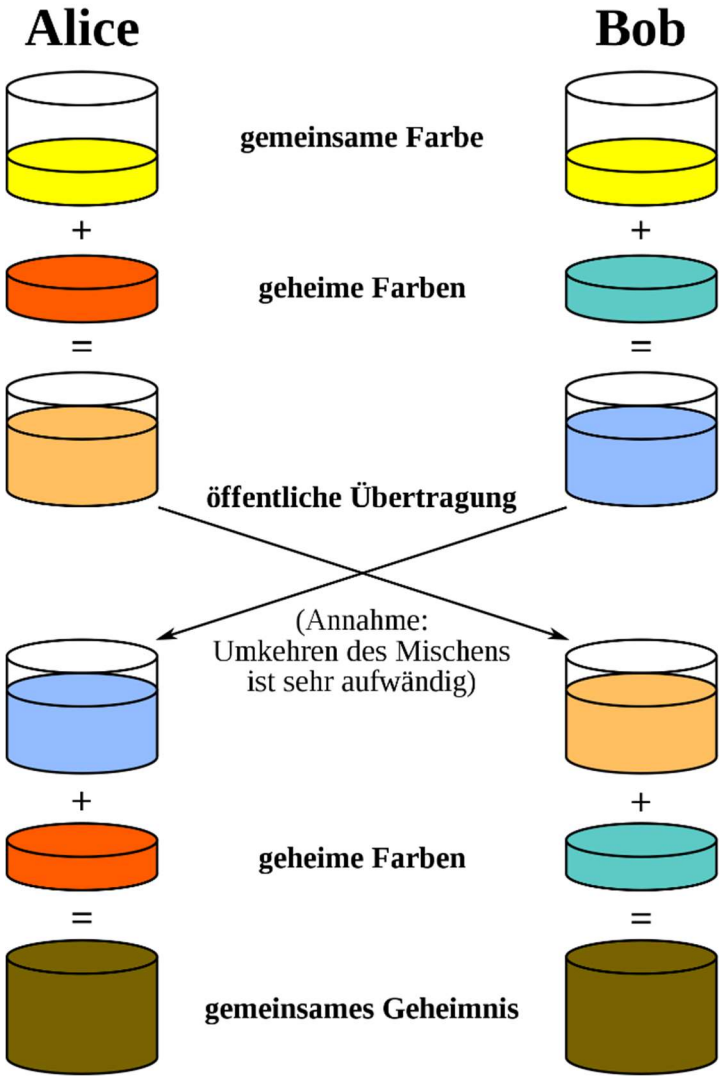

Abbildung 4: Veranschaulichung des DH mittels Farben (Vinck, 2014)

Der Prozess zur Erstellung eines geheimen gemeinsamen Schlüssels, kann mittels Farbenmischen veranschaulicht werden (siehe Abbildung 4). Zuerst müssen sich Absender und Empfänger über ein öffentliches Medium auf eine Startfarbe einigen, in Abbildung 4 ist diese gelb. Dann wählt sowohl der Absender wie auch der Empfänger eine geheime Farbe und mischt sie zu der Startfarbe. In der Abbildung 4 entspricht das der roten und türkisen Farbe. Das entstandene Farbgemisch, in diesem Fall die Farbe Orange, des Absenders kann nun, auch über öffentlich zugängliche Kanäle, zum Empfänger geschickt werden. Dieser wiederum, schickt dem Absender seine entstandene Farbmischung, in Abbildung 4 ist diese blau. Abschließend mischen Absender und Empfänger jeweils ihre geheimen Farben in die gerade erhaltenen Farbmischungen. Nach diesem Prozess haben Absender und Empfänger beide dieselbe Farbe in ihren Töpfen. In der Abbildung 4 entspricht das der Farbe Braun. Auch wenn ein Angreifer alle versendeten Nachrichten und Farbtöpfe zwischen Sender und Empfänger abgefangen hätte, so kann er daraus nicht auf die endgültige geheime gemeinsame Farbe schließen (Singh, 2000).

5.1.2 Theoretische Durchführung

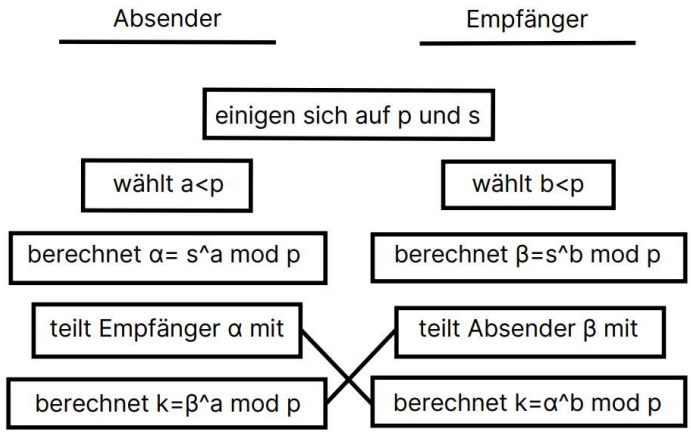

Abbildung 5: Überblick über den Ablauf des DH zwischen Sender und Empfänger

Zuerst einigen sich Sender und Empfänger auf zwei sehr große Zahlen *p* und *s*, wobei *p* für eine Primzahl steht und *s* eine natürliche Zahl größer als 1 und kleiner als *p* ist. Anschließend wählen Sender und Empfänger je eine geheime Zahl, kleiner als *p*, die keinesfalls ausgetauscht wird. Im weiteren Text wird die geheime Zahl des Senders *a* und jene des Empfängers *b* genannt. Nun berechnet der Absender $s^a \ mod \ p$, das erhaltene Ergebnis wird mit α abgekürzt. Der Empfänger berechnet β, indem er $s^b \ mod \ p$ ermittelt. Danach tauschen Sender und Empfänger α und β aus. Dies kann auch über einen öffentlich zugänglichen Kanal geschehen. Hat der Absender β erhalten, so kann er $\beta^a \ mod \ p$ berechnen. Ist die Zahl α beim Empfänger angekommen, so ermittelt er $\alpha^b \ mod \ p$. Absender und Empfänger haben nun gemeinsam $s^{a \times b} \ mod \ p$ über öffentlich zugängliche Kanäle berechnet, ohne vorher Geheimnisse austauschen zu müssen. Die erhaltene gemeinsame geheime Zahl können sie nun als Schlüssel für symmetrische Verschlüsselungsverfahren benutzen (Beutelspacher, Geheimsprachen, 2012), (Ertel & Löhmann, 2020), (Beutelspacher, Kryptologie, 2015).

5.1.3 Praktisches eigenes Beispiel

Der Sender einigt sich mit dem Empfänger auf zwei gemeinsame öffentliche Zahlen p und g. Im Anschluss überlegt sich der Absender eine eigene geheime Zahl a. Danach berechnet der Sender die Zahl α und übermittelt sie dem Empfänger.

$p = 13 \qquad g = 4 \qquad a = 9$

$\alpha = g^a \ mod \ p$

$\alpha = 4^9 \ mod \ 13$

$\alpha = 262.144 \ mod \ 13$

$\alpha = 12$

Wenn der Absender die Zahl β vom Empfänger erhalten hat, kann er die Zahl k berechnen, welche die geheime, gemeinsame Zahl zwischen Sender und Empfänger ist.

$\beta = 10$

$k = \beta^a \ mod \ p$

$k = 10^9 \ mod \ 13$

$k = 1.000.000.000 \ mod \ 13$

$k = 12$

Der Empfänger einigt sich mit dem Sender auf zwei gemeinsame öffentliche Zahlen p und g. Im Anschluss überlegt sich der Empfänger eine eigene geheime Zahl b. Danach berechnet der Empfänger die Zahl β und übermittelt sie dem Absender.

$p = 13 \qquad g = 4 \qquad b = 5$

$\beta = g^b \ mod \ p$

$\beta = 4^5 \ mod \ 13$

$\beta = 1.024 \ mod \ 13$

$\beta = 10$

Wenn der Empfänger die Zahl α vom Empfänger erhalten hat, kann er die Zahl k berechnen, welche die geheime, gemeinsame Zahl zwischen Sender und Empfänger ist.

$\alpha = 12$

$k = \alpha^b \ mod \ p$

$k = 12^5 \ mod \ 13$

$k = 248.832 \ mod \ 13$

$k = 12$

5.1.4 Sicherheit

Damit ein Angreifer auch auf die geheime Zahl kommt, braucht er neben den möglicherweise öffentlich zugänglichen Nachrichten zwischen Sender und Empfänger auch eine der beiden geheimen Zahlen *a* oder *b*. In der Funktion $\alpha = s^a \ mod \ p$ oder $\beta = s^b \ mod \ p$ würde einem Angreifer, wenn er die möglicherweise öffentlich zugänglichen Nachrichten zwischen Sender und Empfänger gelesen hat, in diesen Gleichungen nur mehr *a* oder *b* fehlt. Er muss somit nur eine der beiden Gleichungen nach *a* oder *b* hin auflösen. Doch handelt es sich bei dieser Gleichung um eine Einwegfunktion. Diese Einwegfunktionen haben die Eigenschaft, dass sie in eine Richtung recht einfach durchzuführen sind, ihre Umkehrung aber praktisch unmöglich ist. Die

Einwegfunktion dieses Verfahrens wird durch eine diskrete Exponentialfunktion dargestellt. Ihre Umkehrung ist die diskrete Logarithmusfunktion. Damit die Umkehrung der Einwegfunktion umgangen werden kann, gibt es die Möglichkeit alle möglichen Lösungen auszuprobieren. Dies klingt anfangs recht einfach, nur werden in der Praxis so große Zahlen verwendet, dass es unmöglich ist, die Einwegfunktion auf diese Art zu brechen. Die verwendeten Zahlen, also p, s, a, b, α und β haben normalerweise zwischen 100 und 200 Dezimalstellen. Andere Möglichkeiten diese Verschlüsselungsmethode zu brechen, als die gerade genannten sind nach heutigem Stand der Wissenschaft noch nicht gefunden, können aber, wenn es sie gibt, in der Zukunft entdeckt werden (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010), (Beutelspacher, Geheimsprachen, 2012), (Ertel & Löhmann, 2020), (Schmeh, 2016).

5.2 RSA-Kryptosystem

Das erste asymmetrische Verschlüsselungsverfahren entdeckte eigentlich Clifford Cocks mit Vorarbeit seiner Kollegen beim britischen Geheimdienst. Jedoch wurde die Erfindung abermals unter Verschluss gehalten, da sie für den britischen Geheimdienst zu wichtig war (Singh, 2000). Daraufhin wurde der RSA-Algorithmus 1977 das erste Mal öffentlich von Ronald Rivest, Adi Shamir und Leonard Adleman niedergeschrieben. Er beruht auf dem Faktorisierungsproblem und besteht aus einem privaten Schlüssel und einem öffentlichen Schlüssel. Der Name RSA kommt von den Nachnamen der Erfinder, wobei Ronald Rivest und Adi Shamir immer wieder neue Ideen hatten, welche dann von Leonard Adleman überprüft wurden (Singh, 2000).

5.2.1 Veranschaulichung

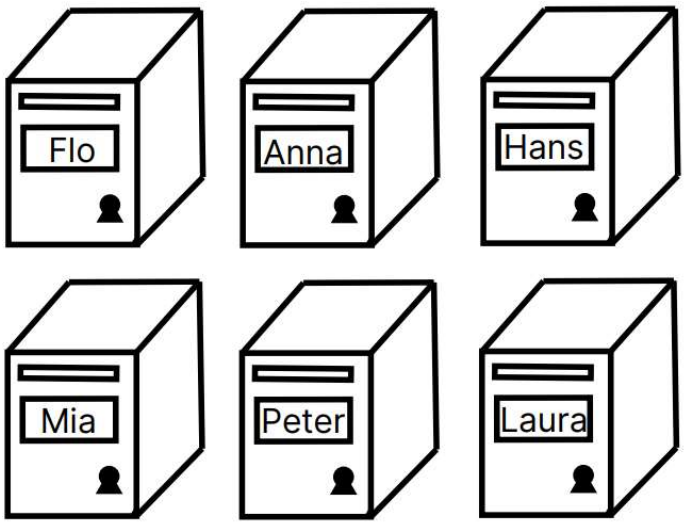

Abbildung 6: Veranschaulichung von RSA mittels Briefkästen

Den ganzen Prozess der Ver- und Entschlüsselung kann anhand von Briefkästen veranschaulicht werden. Eine Person, zum Beispiel Mia (siehe Abbildung 6) hat einen eigenen Briefkasten, zu

dem nur sie selbst einen Schlüssel hat. Will nun jemand eine Nachricht an diese Person, also an Mia schicken, so muss diese Nachricht nur in den richtigen Briefkasten gelegt werden, der das richtige Namensschild trägt. Danach hat nur mehr der Besitzer des Briefkastens, also Mia die Möglichkeit, diesen mit ihrem Schlüssel zu entsperren und die Nachricht zu lesen.

In dieser Veranschaulichung steht der öffentlich zugängliche Briefkasten für den öffentlichen Schlüssel. Das Einwerfen der Nachricht ist mit der Verschlüsselung gleichzusetzten, wobei die Entschlüsselung das Aufsperren des Briefkastens wäre. Nach der Verschlüsselung, dem Einwurf in den Briefkasten, kann nur mehr der Empfänger der Nachricht auf diese zugreifen. Nicht einmal der Absender kann seine eigene Nachricht nochmal lesen (Beutelspacher, Kryptologie, 2015).

5.2.2 Theoretische Durchführung

Für die Schlüsselerzeugung durch den Empfänger, wählt der Empfänger wählt zwei geheime Primzahlen p und q, desto größer, desto sicherer, und berechnet das Produkt N dieser. Anschließend braucht er eine weitere Primzahl e. Der öffentliche Schlüssel des Empfänger ist das Produkt N und die Primzahl e. Anschließend berechnet er seinen privaten Schlüssel, welcher aus dem Produkt N und einer Zahl d besteht. Diese Zahl d berechnet er mit folgender Formel und dem euklidischen Algorithmus: $e \times d = 1(mod(p-1) \times (q-1))$ (Kippenhahn, 2012), (Beutelspacher, Kryptologie, 2015).

Bei der Verschlüsselung der Nachricht durch den Absender, muss der Versender zuerst seine Nachricht in Dezimalzahlen umwandeln. Hierfür eignet sich zum Beispiel der ASCII-Code. Als Nächstes muss der Sender den öffentlichen Schlüssel des Empfängers der Nachricht erfahren. Je nachdem wie lang die Nachricht ist, kann es notwendig sein, diese in deutlich kleinere Zahlenpakete aufzuteilen. Damit die Nachricht M verschlüsselt wird, berechnet der Absender $M^e \ mod \ N$. Hat der Absender mehrere Zahlenpakete, verschlüsselt er alle und reiht sie hintereinander (Beutelspacher, Kryptologie, 2015), (Ertel & Löhmann, 2020).

Für die Entschlüsselung der Nachricht durch den Empfänger trennt der Empfänger sie zuerst wieder auf, da die Nachricht aus mehreren Teilen besteht. Anschließend kann der Empfänger die verschlüsselte Nachricht C im Ganzen oder in Teilen entschlüsseln, indem er $C^d \ mod \ N$ berechnet. Nun muss er die Dezimalzahlen nur noch mit dem ASCII-Code in Buchstaben übersetzen (Beutelspacher, Kryptologie, 2015), (Ertel & Löhmann, 2020).

5.2.3 Praktisches eigenes Beispiel

Der Empfänger wählt für die Schlüsselberechnung zwei unterschiedliche geheime Primzahlen p und q und berechnet das Produkt der beiden Zahlen N.

$p = 11 \qquad q = 13$

$N = p \times q = 11 \times 13 = 143$

Der Empfänger wählt eine weitere Zahl e.

$e = 7$

Nun berechnet der Empfänger noch die Zahl d mit Hilfe des euklidischen Algorithmus.

Euklidischer Algorithmus:

$120 = 7 \times 17 + 1$

$7 = 1 \times 7 + 0$

$1 = 120 \times 1 - 7 \times 17$

$1 = (-7) \times 17 \; mod \; 120$

$e \times d = 1 \; (mod \; (p - 1) \times (q - 1))$

$7d = 1 \; (mod \; (11 - 1) \times (13 - 1))$

$7d = 1 \; mod \; (10 \times 12)$

$7d = 1 \; mod \; 120$

$7d = (-7) \times 17 \; mod \; 120$

$7 \times (-17) \; mod \; 120 = (-7) \times 17 \; mod \; 120$

$d = (-17) \; mod \; 120$

$d = 103$

Somit ist der öffentliche Schlüssel $N = 143$ und $e = 7$ und der private Schlüssel ist $d = 103$. Den öffentlichen Schlüssel muss der Empfänger zum Beispiel auf bestimmten Internetplattformen veröffentlichen.

Der Sender sucht sich, um eine Nachricht verschlüsselt an einen bestimmten Empfänger zu schicken, zuerst den öffentlichen Schlüssel des gewünschten Empfänger heraus. Dieser lautet $N = 143$ und $e = 7$.

Nun muss der Sender die zu verschlüsselnde Nachricht mittels ASCII-Tabelle in Dezimalzahlen übersetzten (siehe Abbildung 7). Hierbei erhält er für einen Buchstaben immer eine Zahl unter Hundert.

Um die geheime Nachricht C sicher zu verschlüsseln, sollte der Absender immer mindestens zwei Buchstaben zusammengefasst verschlüsseln, um Angreifern nicht die Möglichkeit einer Häufigkeitsanalyse zu geben. Bei einer ungeraden Anzahl an Buchstaben kann der Sender ein X

an die Nachricht anhängen. Dafür müsste aber N größer als die einzelnen Teile der verschlüsselten Nachricht V sein. Da Zahlen hoch einer Potenz im tausender Bereich zu groß für eine händische Berechnung werden, wurde in diesem Beispiel dieser Sicherheitsaspekt außer Acht gelassen und die jeweils einzelnen Buchstaben verschlüsselt.

G	E	H	E	I	M	N	I	S
71	69	72	69	73	77	78	73	83

Abbildung 7: Umwandlung der Buchstaben mittels ASCII-Tabelle in Dezimalzahlen

$C^e \ mod \ N = V$

$71^7 \ mod \ 143 = 9095120158391 \bmod 143 = 124$

$69^7 \ mod \ 143 = 7446353252589 \bmod 143 = 108$

$72^7 \ mod \ 143 = 10030613004288 \bmod 143 = 19$

$73^7 \ mod \ 143 = 11047398519097 \bmod 143 = 83$

$77^7 \ mod \ 143 = 16048523266853 \bmod 143 = 77$

$78^7 \ mod \ 143 = 17565568854912 \bmod 143 = 78$

$83^7 \ mod \ 143 = 27136050989627 \bmod 143 = 8$

Schlussendlich muss der Absender die verschlüsselte Nachricht wieder zusammensetzen und dem Empfänger zukommen lassen. Die zusammengesetzte verschlüsselte Nachricht lautet $124108019108083077078083008$.

Der Empfänger erhält vom Absender die verschlüsselte Nachricht $124108019108083077078083008$. Er teilt sie wieder in Teile auf und entschlüsselt sie mit dem eigenen privaten Schlüssel.

$C = V^d \ mod \ N$

$124^{10} \ mod \ 143 = 859442550649180389376 \bmod 143 = 56$

$124^3 \bmod 143 = 1906624 \bmod 143 = 5$

$124^{103} \bmod 143 = [124^3 \bmod 143 \times (124^{10} \bmod 143)^{10}] \bmod 143 = 5 \times 56^{10} \bmod 143$

$= 1516527445480570880 \bmod 143 = 71$

$108^{10} \bmod 143 = 215892499727278669824 \bmod 143 = 100$

$108^{3} \bmod 143 = 1259712 \bmod 143 = 25$

$108^{103} \bmod 143 = [108^{3} \bmod 143 \times (108^{10} \bmod 143)^{10}] \bmod 143$

$= 25 \times 100^{10} \bmod 143 = 2500000000000000000000 \bmod 143 = 69$

$19^{10} \bmod 143 = 6131066257801 \bmod 143 = 56$

$19^{3} \bmod 143 = 6859 \bmod 143 = 138$

$19^{103} \bmod 143 = [19^{3} \bmod 143 \times (19^{10} \bmod 143)^{10}] \bmod 143 = 138 \times 56^{10} \bmod 143$

$= 41856157495263756288 \bmod 143 = 72$

$108^{10} \bmod 143 = 215892499727278669824 \bmod 143 = 100$

$108^{3} \bmod 143 = 1259712 \bmod 143 = 25$

$108^{103} \bmod 143 = [108^{3} \bmod 143 \times (108^{10} \bmod 143)^{10}] \bmod 143$

$= 25 \times 100^{10} \bmod 143 = 2500000000000000000000 \bmod 143 = 69$

$83^{10} \bmod 143 = 15516041187205853449 \bmod 143 = 12$

$83^{3} \bmod 143 = 571787 \bmod 143 = 73$

$83^{103} \bmod 143 = [83^{3} \bmod 143 \times (83^{10} \bmod 143)^{10}] \bmod 143 = 73 \times 12^{10} \bmod 143$

$= 4519967588352 \bmod 143 = 73$

$77^{10} \bmod 143 = 7326680472586649 \bmod 143 = 66$

$77^{3} \bmod 143 = 456533 \bmod 143 = 77$

$77^{103} \bmod 143 = [77^{3} \bmod 143 \times (77^{10} \bmod 143)^{10}] \bmod 143 = 77 \times 66^{10} \bmod 143$

$= 120761939830131274752 \bmod 143 = 77$

$78^{10} \bmod 143 = 8335775831236199424 \bmod 143 = 78$

$78^{3} \bmod 143 = 474552 \bmod 143 = 78$

$78^{103} \bmod 143 = [78^{3} \bmod 143 \times (78^{10} \bmod 143)^{10}] \bmod 143 = 78 \times 78^{10} \bmod 143$

$= 650190514836423555072 \bmod 143 = 78$

$83^{10} \bmod 143 = 15516041187205853449 \bmod 143 = 12$

$83^{3} \bmod 143 = 571787 \bmod 143 = 73$

$83^{103} \bmod 143 = [83^{3} \bmod 143 \times (83^{10} \bmod 143)^{10}] \bmod 143 = 73 \times 12^{10} \bmod 143$

$= 4519967588352 \bmod 143 = 73$

$8^{10} \, mod \, 143 = 1073741824 \bmod 143 = 12$

$8^{3} \bmod 143 = 512 \bmod 143 = 83$

$8^{103} \bmod 143 = [8^{3} \bmod 143 \times (8^{10} \bmod 143)^{10}] \bmod 143 = 83 \times 12^{10} \bmod 143$

$= 5139141230592 \bmod 143 = 83$

Abschließend muss der Empfänger die entschlüsselten Zahlen nur mehr mittels ASCII-Code in Buchstaben übersetzten (siehe Abbildung 8).

71	69	72	69	73	77	78	73	83
G	E	H	E	I	M	N	I	S

Abbildung 8: Umwandlung der Dezimalzahlen mittels ASCII-Tabelle in Buchstaben

5.2.4 Sicherheit

Damit eine mit RSA verschlüsselte Nachricht, ohne privaten Schlüssel, entschlüsselt wird, muss ein Angreifer versuchen auf die Zahl d zu kommen. Doch um diese zu berechnen, muss er die Primzahlen *p* und *q* kennen, mit welchen *N* berechnet wurde. Allerdings ist einem Angreifer nur das Produkt *N* aus den beiden Primzahlen bekannt. Ein Angreifer müsste N faktorisieren, um p und q zu erhalten. Dies ist jedoch bedeutend schwieriger, als die zwei Primzahlen miteinander zu multiplizieren. Denn um *N* in seine Primfaktoren zu zerlegen, müsste ein Angreifer jede Primzahl einzeln ausprobieren. Hierfür gibt es Algorithmen, aber können diese die Arbeit nur gering beschleunigen. Außerdem werden mittlerweile so große Primzahlen verwendet, dass selbst wenn sich die ganze Welt mit all ihren Computern es sich zur einzigen Aufgabe macht, dieses Produkt zu faktorisieren, es mehrere tausend Jahre brauchen würde. Stand 2015 wurden erst sehr wenige 512 und 768 Bit lange Zahlen faktorisiert, weshalb empfohlen wird Zahlen mit einer Länge von

1024 oder sogar 2048 Bits zu verwenden (Beutelspacher, Schwenk, & Wolfenstetter, Moderne Verfahren der Kryptographie, 2015), (Kippenhahn, 2012), (Beutelspacher, Geheimsprachen, 2012), (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010), (Ertel & Löhmann, 2020).

5.2.5 Anwendung

Damit eine bestimmte Sicherheit garantiert werden kann, müssen die Primzahlen sehr groß sein. Mit steigender Länge der Zahlen steigt aber auch der Aufwand, um eine Nachricht zu ver- und entschlüsseln, weil die Berechnungen Potenzen und Modulo enthalten. Außerdem ist es sehr schwierig den RSA-Algorithmus halbwegs vernünftig in ein Computerprogramm zu implementieren. Somit wird das RSA-Kryptosystem häufig für elektronische Signaturen und das Schlüsselmanagement verwendet, jedoch nicht für die Verschlüsselung von Nachrichten.

Damit dennoch die Sicherheit asymmetrischer Systeme, auch für die Verschlüsselung von Nachrichten, genutzt werden kann, werden sie mit symmetrischen Verfahren kombiniert. Kombinationen aus asymmetrischen- und symmetrischen Verfahren werden hybride Verfahren genannt. Sie vereinen die Vorteile asymmetrischer Systeme - die Sicherheit, mit jenen der symmetrischen Verfahren - die Schnelligkeit. Zusätzlich gibt es die Möglichkeit mit öffentlichen Schlüsseln zu arbeiten, womit der geheime Schlüsselaustausch mit jedem einzelnen Konversationspartner nicht notwendig ist. Ein Beispiel für ein hybrides Verfahren, welches auch als ein Computerprogramm existiert, wäre Pretty Good Privacy (Siehe Kapitel 5.4) (Ertel & Löhmann, 2020).

5.2.6 Signieren

Wie schon oben erwähnt, ist ein Anwendungsbereich von RSA die elektronische Signatur. Hierbei verschlüsselt der Absender die Nachricht zuerst mit seinem privaten Schlüssel und dann mit dem öffentlichen Schlüssel vom Empfänger. Dabei ist die Reihenfolge sehr wichtig, um sicher zu gehen, dass die Signatur von Unbefugten nicht verändert werden kann. Der Empfänger wiederum entschlüsselt die Nachricht zuerst mit seinem privaten Schlüssel und dann mit dem öffentlichen Schlüssel des Absenders. Da der Empfänger davon ausgeht, dass nur der Versender seinen eigenen privaten Schlüssel kennt, muss die Nachricht direkt vom Absender kommen. Oft wird aber nicht die ganze Nachricht signiert, da dies recht aufwendig ist. Stattdessen wird ein Hashwert der Nachricht berechnet und anschließend nur dieser mit der Signatur versehen. Wobei ein Hashwert das Ergebnis einer Streuwertfunktion ist, welche eine bestimmte Art von Funktionen ist, mit der Eigenschaft vielen Eingabewerten weniger Ausgabewerte zuzuordnen. Die Hashwerte sind meist bestimmte skalare Werte aus den natürlichen Zahlen. Dadurch, dass es weniger Ausgabewerte als Eingabewerte gibt, ist es unvermeidbar, dass Kollisionen auftreten, also mindestens zwei Eingabewerte dasselbe Ergebnis haben. Eine gute Streuwertfunktion jedoch

vermeidet diese Kollisionen für eine bestimmte Eingabemenge (Beutelspacher, Kryptologie, 2015), (Schmeh, 2016).

5.3 Elliptische-Kurven-Kryptografie

Damit sich die Schlüssellänge asymmetrischer Verschlüsselungsverfahren verkürzt, kamen Neal Koblitz und Victor S. Miller 1985 unabhängig voneinander auf die Idee, elliptische Kurven zu verwenden. Im Folgenden wird die Kryptografie mit elliptischen Kurven am Beispiel des Diffie-Hellman-Merkte-Schlüsseltausch, kurz ECDH, beschrieben. Hierbei wird die Schwierigkeit des Lösens des diskreten Logarithmus großer Zahlen modulo großer Primzahlen durch die Schwierigkeit des Lösens des diskreten Logarithmus elliptischer Kurven modulo großer Primzahlen ersetzt (Beutelspacher, Kryptologie, 2015), (Wong, 2023).

5.3.1 Grundlagen zu elliptischen Kurven

Elliptische Kuren können mithilfe der Gleichung $y^2 = ax^3 + bx^2 + cx + d$ definiert werden. Diese speziellen Kurven haben die Eigenschaft, dass jede Gerade, die nicht parallel zur y-Achse ist, eine oder drei Schnittpunkte mit der Kurve besitzt.

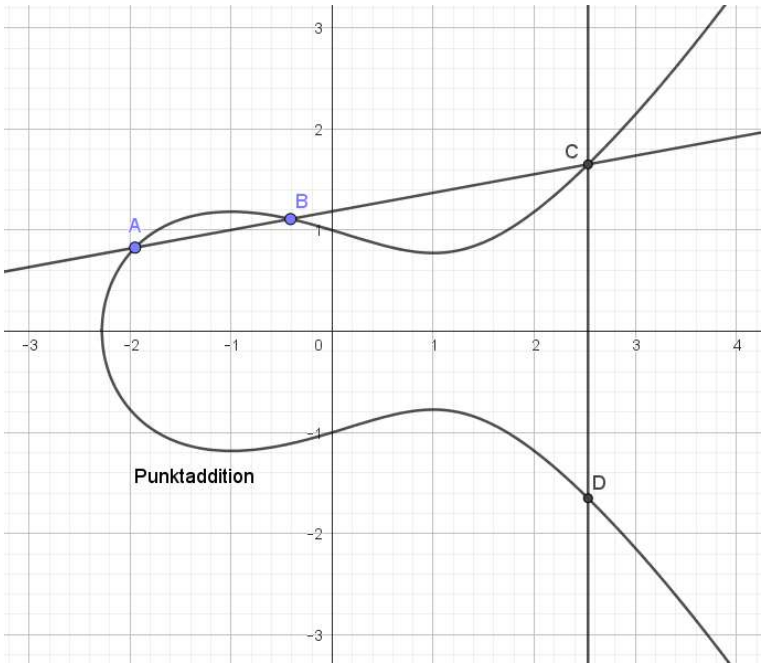

Abbildung 9: Punktaddition auf elliptischen Kurven

Werden zwei Punkte *A* und *B* miteinander addiert, so wird eine Linie durch beide Punkte gezogen. Diese Linie schneidet die Kurve in einem weiteren Punkt *C*. Nun wird von dem Punkt *C* ausgehend eine senkrechte Linie, welche die Kurve in einem weiteren Punkt *D* schneidet, gezeichnet. Dieser Punkt *D* ist das Ergebnis der Addition von Punkt *A* und Punkt *B* (siehe Abbildung 9).

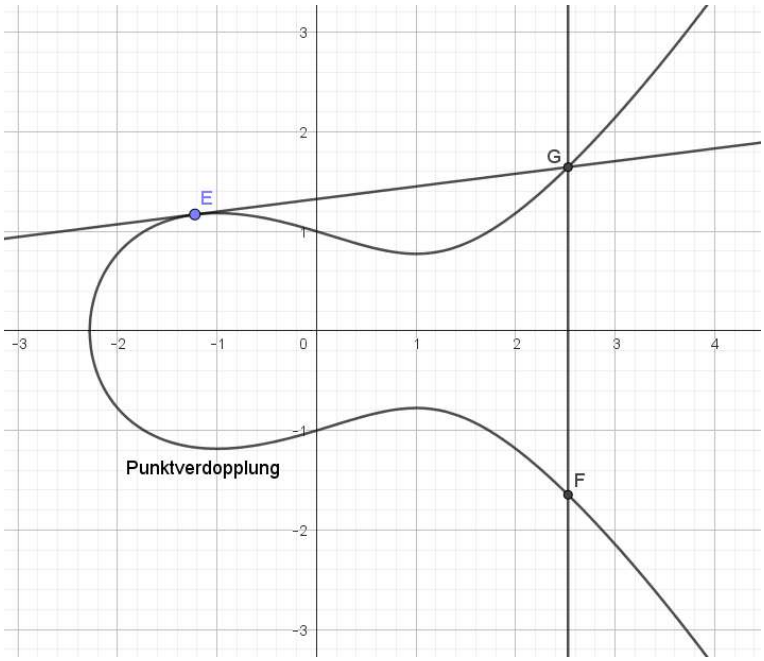

Abbildung 10: Punktverdopplung auf elliptischen Kurven

Wird ein Punkt *E* mit sich selbst addiert, also verdoppelt, so wird die Tangente zu diesem Punkt gezeichnet. Anschließend wird eine senkrechte Linie ausgehend von einem anderen Schnittpunkt der Tangente mit der Kurve gezeichnet. Der Punkt *F*, an welchem die senkrechte Linie die Kurve und aber nicht die Tangente schneidet, ist das Ergebnis, der Verdopplung von Punkt *E* (siehe Abbildung 10).

Hat eine gezogene Linie oder Tangente keine neuen Schnittpunkte mit der Kurve, so endet diese Addition im Unendlichkeitspunkt. Er wird normalerweise mit einem großen *O* bezeichnet und verhält sich wie eine Null (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010), (Schmeh, 2016), (Wong, 2023).

5.3.2 Theoretische Durchführung

Damit mittels Elliptic-Curve-Diffie-Hellman-Merkle-Schlüsseltausch, kurz ECDH, ein geheimer gemeinsamer Schlüssel erzeugt werden kann, müssen sich zuerst Sender und Empfänger auf eine elliptische Kurvengleichung, eine große Primzahl *P* und einen Punkt *B* auf der Kurve einigen. Dies kann, wie auch bei dem normalen Schlüsseltausch, über öffentlich zugängliche Kanäle erfolgen. Danach generieren Sender und Empfänger jeweils eine große geheime Zufallszahl mittels eines Zufallsgenerators. Die folgenden Vervielfachungen des Punktes *B* geschehen nach den im Kapitel 5.3.1 vorgestellten Regeln zur Verdoppelungen des Punktes *E*. Im Anschluss wird jene Zufallszahl vom Sender als *s* und jene vom Empfänger als *e* bezeichnet. Dann berechnet der Absender $s \times B \; mod \; P$ und übermittelt sein Ergebnis *x* dem Empfänger. Dieser wiederum berechnet $e \times B \; mod \; P$ und sendet das Resultat *y* dem Absender. Hat der Sender *y* erhalten, so ermittelt er $s \times y \; mod \; P = g$. Hat der Empfänger *x* ebenfalls erhalten, so berechnet er $e \times x \; mod \; P = g$. Absender und Empfänger haben nun gemeinsam $s \times e \times B \; mod \; P$ über

öffentlich zugängliche Kanäle berechnet, ohne vorher Geheimnisse austauschen zu müssen. Die erhaltene gemeinsame geheime Zahl *g* können sie nun, wie beim normalen Diffie-Hellman-Merkle-Schlüsseltausch, als geheimen gemeinsamen Schlüssel für symmetrische Verschlüsselungsverfahren benutzen. Statt der, wie in dieser Beschreibung erklärten, Multiplikation von *s* oder *e* mit dem Punkt *B* oder den Resultaten *x* und *y*, kann *B*, *x* und *y* auch durch *s* oder *e* potenziert werden (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010), (Ertel & Löhmann, 2020).

5.3.3 Vorteile

Aufgrund der Verwendung der elliptischen Kurven funktionieren die stärksten bekannten Angriffe auf den normalen Diffie-Hellman-Merkle-Schlüsseltausch nicht. Somit können die Zahlen, wenn ECDH verwendet wird, um einiges kleiner sein, als wenn nur DH verwendet wird. Sind die Zahlen in DH 2.048 Bits lang, werden mit ECDH für dieselbe Sicherheit nur 256 Bit lange Zahlen benötigt (Wong, 2023), (Schmeh, 2016).

5.4 Pretty Good Privacy

Pretty Good Privacy oder auch kurz PGP, ist eine Software, welche im Juni 1991 von Phil Zimmermann jedem kostenlos im Internet zur Verfügung gestellt wurde. Die Verschlüsselung dieser Software ist hybrid, das heißt sie verwendet sowohl symmetrische wie auch asymmetrische Verschlüsselungsverfahren. Damit vereint sie die Vorteile beider Systeme, das Verfahren ist sehr schnell, aber auch sicher und hat das Problem des geheimen Schlüsselaustausch nicht. Außerdem ist es möglich mit diesem Programm seine Nachricht nicht nur Verschlüsseln, sondern auch zu signieren und seine eigenen Schlüssel zu erzeugen. Bei all den Funktionen war es Zimmermann immer wichtig, die Verwendung des Programmes so einfach wie möglich zu gestalten, sodass die Software schlussendlich alles automatisch erledigt (Beutelspacher, Geheimsprachen, 2012), (Singh, 2000).

5.4.1 Schlüsselerzeugung

Damit Nachrichten asymmetrisch verschlüsselt werden können, braucht es einen öffentlichen und einen privaten Schlüssel. Diese sind normalerweise nicht so einfach zu berechnen, da hierfür mit zwei sehr großen Primzahlen gerechnet werden muss. In PGP wird das dem Nutzer so weit abgenommen, dass dieser nur ein paar Sekunden mit seiner Maus über den Bildschirm fahren muss. Aus diesen Bewegungen ergibt sich ein Zufallsfaktor, womit durch Berechnungen jedem Benutzer ein eigenes Schlüsselpaar ermittelt wird (Singh, 2000).

5.4.2 Ver- und Entschlüsselung ohne Signatur

Die Software verschlüsselt eine Nachricht zuerst mit einem symmetrischen Verfahren namens IDEA. Anschließend sucht sie den öffentlichen Schlüssel des Empfängers und verschlüsselt damit

den Schlüssel des symmetrischen Verfahrens asymmetrisch. Nun wird die Nachricht verschickt und das Programm des Empfängers entschlüsselt zuerst mit dem privaten asymmetrischen Schlüssel den symmetrischen Schlüssel der Nachricht. Anschließend entschlüsselt es die Nachricht mit dem gerade zuvor entschlüsselten symmetrischen Schlüssel und zeigt sie auf dem Bildschirm an (Singh, 2000), (Küsters & Wilke, 2011).

5.4.3 Ver- und Entschlüsselung mit Signatur

Die Software verschlüsselt eine Nachricht zuerst mit einem symmetrischen Verfahren namens IDEA. Dann verschlüsselt das Programm des Absenders die Nachricht inklusive des symmetrischen Schlüssels mit dem eigenen privaten Schlüssel asymmetrisch. Anschließend sucht es den öffentlichen asymmetrischen Schlüssel des Empfängers und verschlüsselt damit nochmals die ganze Nachricht. Nun wird die Nachricht verschickt und das Programm des Empfängers entschlüsselt zuerst mit dem eigenen privaten asymmetrischen Schlüssel und darauffolgend mit dem öffentlichen asymmetrischen Schlüssel des Absenders. Abschließend wird die Nachricht mit dem gerade entschlüsselten symmetrischen Schlüssel entschlüsselt und angezeigt (Singh, 2000), (Wong, 2023).

5.4.4 Anwendung

Durch ihre leichte Anwendung findet die freizugängliche Software, welche Anfangs auf einem Bulletin Board im Usenet ausgehängt war, breite Verwendung auf der ganzen Welt. Das Programm wurde zum Beispiel von Menschenrechts- und Widerstandsgruppen aber auch in der breiten Öffentlichkeit verwendet (Singh, 2000).

6. Weitere Entwicklung – (Post-)Quantenkryptografie

Früher wurden in der Kryptografie einfache mathematische Techniken wie Substitutionen und Transpositionen verwendet. Außerdem beruhte die Sicherheit dieser kryptografischen Methoden darauf, dass die verwendeten Verschlüsselungsverfahren nicht allgemein bekannt waren. Trotzdem gelang es den damaligen Kryptoanalytikern viele Kryptografieverfahren zu brechen. Deswegen mussten die Kryptografen immer neue Methoden erfinden und bekannte Verfahren weiterentwickeln. Heute gibt es den Standard, dass die verwendeten Verfahren öffentlich bekannt sind, dadurch soll sichergestellt werden, dass die Sicherheit der Verfahren nicht auf Unwissenheit, sondern auf der Komplexität der verwendeten Methoden beruht. Asymmetrische Kryptografieverfahren beruhen zusätzlich oft auf sehr schweren mathematischen Problemen, wie zum Beispiel dem Faktorisieren von sehr großen Zahlen oder dem Berechnen des diskreten Logarithmus von sehr großen Zahlen. Wie dies auf die Spitze getrieben wird, erkennt man anhand folgender beider Zitate.

> Die Kryptographen neigen ein wenig zu Paranoia und stellen sich gerne die größten anzunehmenden Katastrophen vor, etwa eine weltweite Verschwörung mit dem Ziel, ihre Verschlüsselungen zu knacken. (Singh, 2000, S. 335)

> Inzwischen werden Nachrichten mit hinreichend großen Werten […] verschlüsselt, so daß alle Computer des Planeten länger brauchen würden, als das Universum alt ist, um die Verschlüsselung zu knacken. (Singh, 2000, S. 337)

6.1 Grundlagen

Elementarteilchen können laut Quantenmechanik gleichzeitig mehrere Zustände annehmen, die sich eigentlich gegenseitig ausschließen. Dieser Zustand wird Superposition genannt und wird in dem Moment einer Messung des Zustandes des Elementarteilchens beendet. Dabei ist es nur möglich Wahrscheinlichkeiten anzugeben, da es nicht vorhersehbar ist, welcher Zustand schlussendlich gemessen wird. Beispielsweise kann ein Elektron innerhalb eines Magnetfeldes zwei verschiedene Spins haben. Befindet sich das Elektron in Superposition, wird erst bei der Messung der wahre Spin festgelegt.

Zwischen zwei oder mehr Elementarteilchen kann auch der Zustand der Verschränkung bestehen. Dabei beeinflusst das Ergebnis einer Messung einer Eigenschaft eines Teilchens die Eigenschaften der jeweils anderen Elementarteilchen. Dies ist unabhängig von dem Ort, an denen sich die Elementarteilchen befinden.

Werden die Phänomene Superposition und Verschränkung am Beispiel eines Elektrons miteinander kombiniert, ist es möglich den Spin eines Elektrons, welches in Superposition war, zu kennen, ohne die Eigenschaften dieses Elektrons zu messen. Wird der Spin von einem

verschränkten sich in Superposition befindlichen Elektrons gemessen, so ist der Spin des damit verschränkten Elektrons genau gegensätzlich zu dem gemessenen Spin.

Photonen schwingen normal auf ihre Ausbreitungsrichtung in mehrere bestimmte Richtungen. Diese Eigenschaft von Photonen wird Polarisation genannt. Mit Polarisationsfiltern können Photonen in bestimmte Richtungen polarisiert und deren Polarisation gemessen werden. Doch wird die Durchlasswahrscheinlichkeit bei einem nicht mit der wahren Schwingungsrichtung des Photons ausgerichteten Polarisationsfilter nur verringert, bleibt aber trotzdem größer null (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010), (Wong, 2023), (Schmeh, 2016).

6.2 Quantenschüsselverteilung

Basierend auf den Überlegungen von Stephen Wiesner über die Möglichkeit eines Quantengeldes, begannen Charles Bennett und Gilles Brassard ab 1984 an einer Theorie über Quantenkryptografie zu arbeiten. Ihre Idee bewiesen sie 1988 in einem Experiment, in welchem zwei Computer 30 cm voneinander entfernt mit dieser Methode Nachrichten ver- und entschlüsselten. Mittlerweile ist es Forschern unter speziellen Bedingungen gelungen über weitaus größere Distanzen miteinander zu kommunizieren.

Bei der Quantenkryptografie wird polarisiertes Licht, also polarisierte Photonen verwendet, wobei als Polarisationsfilter ein doppelbrechender Kristall dient. Photonen können horizontal (0°) und vertikal (90°) aber auch schräg, das heißt mit einem Winkel von 45° oder 135°, polarisiert werden. Der doppelbrechende Kristall kann aber nur jeweils zwischen 0° und 90° oder 45° und 135° genau unterscheiden, indem er passend polarisierte Photonen geradlinig durchlässt und darauf normal polarisierte Photonen in eine bestimmte Richtung ablenkt. Dies beruht auf der Heisenbergschen Unschärferelation, nachdem bei einem Photon die Polarisation entweder in gerader Richtung (0° und 90°) oder in schräger Richtung (45° und 135°) gemessen werden kann. Die jeweils schräg polarisierten Photonen polarisiert er mit einer Wahrscheinlichkeit von 50 % entweder vertikal oder horizontal um oder lenkt sie ab. Somit muss vor jeder Messung entschieden werden, ob zwischen 0° und 90° oder zwischen 45° und 135° polarisierten Photonen genau unterschieden werden soll.

Wollen nun Sender und Empfänger eine geheime gemeinsame Information erzeugen, so muss der Sender zufällig polarisierte Photonen erzeugen und diese dem Empfänger übermitteln. Dieser wählt für jedes eintreffende Photon eine Stellung seines Filter. Er misst also entweder die gerade Richtung (0° und 90°) oder in schräger Richtung (45° und 135°), aber nie beide gleichzeitig. Anschließend teilt der Empfänger dem Sender über einen öffentlich zugänglichen Kanal mit, bei welchem Photon er welche Stellung des Filters verwendet hat. Daraufhin meldet der Sender ihm die richtig gemessenen Photonen. Wird dieser Schritt übersprungen, sind nur ca. 50 % der

gemessenen Photonen richtig. Sender und Empfänger müssen sich jetzt nur mehr einigen, wie sie aus den gemessenen Daten eine Bitfolge erzeugen. Beispielsweise können sie 0° und 135°, als 0 beziehungsweise 45° und 90°, als 1 definieren.

Indem Sender und Empfänger ein paar Messergebnisse vergleichen, können sie einen externen Lauschangriff ausschließen, da jede Messung durch einen Angreifer mit einem falsch ausgerichteten Filter die Polarisation verändern würde. Somit wären ungefähr 25% der vereinbarten Bits zwischen Sender und Empfänger fehlerhaft (Beutelspacher, Neumann, & Schwarzpaul, Kryptografie in Theorie und Praxis, 2010), (Singh, 2000).

6.3 Post-Quanten-Kryptografie

Quantencomputer bestehen aus Elementarteilchen, welche verschränkt werden können und in Superposition treten. Ein System, in dem zum Beispiel ein Elektron zwei Zustände annehmen kann, wird Quantenbit oder kurz Qubit genannt. Sind Qubits miteinander verschränkt, so können Quantencomputer parallele Funktionen gleichzeitig berechnen. Dies ist ein großer Fortschritt gegenüber herkömmlichen Computer, da diese immer nur eine Funktion nach der anderen ausführen können. Somit lassen sich bestimmte Aufgaben auf einem Quantencomputer deutlich effizienter durchführen als auf einem „normalen" Computer.

Einer dieser Aufgaben, die auf Quantencomputern deutlich schneller zu lösen sind, ist das Faktorisieren von Zahlen. Für die Ausführung dieser Aufgabe schrieb 1994 Peter Shor einen Algorithmus für Quantencomputer. Mit diesem Algorithmus wird das Faktorisieren von Zahlen mittels Computer zu einem effizient zu lösendem Problem. Das Faktorisieren von großen Zahlen ist nicht das einzige mathematische Problem, welches mit Hilfe von Quantencomputern effizient zu lösen ist. Shor zeigte auch, dass das Problem des diskreten Logarithmus mit einem ähnlichen Algorithmus, wie jenem für das Faktorisieren von Zahlen, effizient zu lösen ist. Somit gibt es seit mehr als 20 Jahren einen Algorithmus um fast alle heute bekannten Public-Key-Kryptosysteme zu brechen und es fehlt nur mehr ein funktionsfähiger Quantencomputer für die Verwendung des Algorithmus.

Nach der Entdeckung von Algorithmen für Quantencomputer, mit denen fast jede asymmetrische Verschlüsselung gebrochen werden kann, begann die Suche nach quantenresistenten asymmetrischen Verschlüsselungsmethoden. Zurzeit sind fünf Familien von Verschlüsselungssystemen bekannt, wobei zwei davon nur für das quantenresistente Signieren von Nachrichten zu verwenden sind.

Ein Verfahren, der SI-Schlüsselaustausch, kann für den quantenresistenten Schlüsselaustausch verwendet werden, wobei es wichtig ist eine sehr große Schlüssellänge zu wählen, womit dieses Verfahren sehr aufwendig wird. Ein anderes Verfahren ermöglicht das quantenresistente

Verschlüsseln und Entschlüsseln, wobei es auch sehr große Schlüssellängen braucht und damit sehr aufwendig ist.

Gitterbasierte Verfahren können zum Verschlüsseln und Signieren verwendet werden. Ein Vertreter dieser Verfahren ist NTRU, wobei zwischen NTRU-Encrypt für Ver- und Entschlüsselung und NTRU-Sign für das Signieren von Nachrichten unterschieden wird. NTRU bietet oder Vorteil, bei gleicher Schlüssellänge um einiges schneller als RSA zu sein. Die einzigen Nachteile von NTRU sind seine Komplexität und die Wichtigkeit die richtigen Parameter zu wählen (Wong, 2023), (Schmeh, 2016).

7. Fazit

Zusammenfassend lässt sich sagen, dass sich Kryptografie seit den ersten Verwendungen immer weiterentwickelt hat und sich auch immer weiterentwickeln wird. Einerseits wird der Gegner immer besser und schneller beim Brechen von kryptografischen Verfahren oder heutzutage die Rechenleistung von Computern immer größer. Andererseits werden neue Methoden gefunden, um kryptografische Verschlüsselungsmethoden zu entziffern. Somit werden bei bestehenden kryptografischen Systemen die Schlüssellängen und somit der mögliche Schlüsselraum erhöht. Zusätzlich werden neue Verschlüsselungsmethoden erfunden, welche auf neuen mathematischen Techniken und Problemen beruhen. Dieses Spiel zwischen Kryptologie und Kryptoanalyse wird ewig weitergehen, bis die Quantenschlüsselverteilung praktisch umgesetzt werden kann. Bis die Quantenkryptografie in der Praxis funktioniert, gibt es viele verschiedene Verschlüsselungsmethoden, welche alle ein bestimmtes Maß an Sicherheit bieten, womit auch jeweils bestimmte Vor- und Nachteile verbunden sind. Das Verfahren mit der größten möglichen Sicherheit ist das One-Time-Pad. Jedoch muss bei diesem Verfahren der Schlüssel, welcher genausolang sein muss wie die zu verschlüsselnde Nachricht, geheim zwischen Sender und Empfänger für jede Nachricht einzeln ausgetauscht werden. Anstatt One-Time-Pad gibt es noch einige andere Verschlüsselungsmethoden, welche nicht ganz so sicher, aber dafür um einiges leichter in der Durchführung sind. Beispiele hierfür sind vor allem Vertreter der asymmetrischen Kryptografie wie das RSA-Kryptosystem. Asymmetrische Verfahren beruhen auf bis jetzt noch nicht gelösten mathematischen Problemen, das RSA-Verfahren beruht auf dem Problem des effizienten Faktorisierens von sehr großen Zahlen. Somit gelten asymmetrische Methoden als sicher, solange das jeweilige Problem nicht gelöst ist oder es keine Quantencomputer gibt, welche die mathematischen Probleme effizient lösen können. Ein großer Vorteil von asymmetrischen Kryptografieverfahren gegenüber von One-Time-Pad ist die geringere Schlüssellänge. Außerdem wird nicht für jede Nachricht ein neuer Schlüssel benötigt, sondern jede Person hat einen eigenen öffentlichen und privaten Schlüssel. Somit wird auch das Problem vom geheimen Schlüsselaustausch zwischen Sender und Empfänger umgangen. Abschließend lässt sich also sagen, dass es sehr viele verschiedene Kryptografieverfahren gibt, wobei jede Verschlüsselungsmethode andere Vor- und Nachteile hat. Somit ist es essenziell, für die Wahl des richtigen Verfahrens viele verschiedene Verschlüsselungsmethoden bezüglich ihrer Eigenschaften und des Verwendungszweckes abzuwägen.

Da der Umfang der existierenden Verschlüsselungsmethoden den Rahmen dieser Arbeit weitaus überstiegen hätte, wurde in dieser Arbeit nur eine Auswahl der wichtigsten Kryptografieverfahren vorgestellt. Wobei es anfänglich sehr schwierig war, mich auf einige wenige Verschlüsselungsmethoden zu beschränken. Eine andere Herausforderung war die Erstellung und Berechnung von eigenen Beispielen. Alles in allem war es sehr faszinierend auf welchen einfachen mathematischen Techniken die Kryptografie beruht.

8. Literaturverzeichnis

Bauer, F. (2000). *Entzifferte Geheimnisse: Methoden und Maximen der Kryptologie.* Berlin und Heidelberg: Springer.

Beutelspacher, A. (2012). *Geheimsprachen* (5. aktualisierte Ausg.). München: C.H. Beck Verlag.

Beutelspacher, A. (2015). *Kryoptologie: Eine Einführung in die Wissenschaft vom Verschlüsseln, Verbergen und Verheimlichen* (10. aktualisierte Ausg.). Wiesbaden: Springer Spektrum.

Beutelspacher, A., Neumann, H. B., & Schwarzpaul, T. (2010). *Kryptografie in Theorie und Praxis: Mathematische Grundlagen für Internetsicherheit, Mobilfunk und elektronisches Geld.* (2. überarbeitete Ausg.). Wiesbaden: Vieweg+Teubner.

Beutelspacher, A., Schwenk, J., & Wolfenstetter, K.-D. (2015). *Moderne Verfahren der Kryptographie: Von RSA zu Zero-Knowledge.* (8. überarbeitete Ausg.). Wiesbaden: Springer Spektrum.

Buchmann, J. (2010). *Einführung in die Krypthografie* (5. Ausg.). Berlin und Heidelberg: Springer-Verlag.

Ertel, W., & Löhmann, E. (2020). *Angewandte Kryptographie* (6. aktualisierte Ausg.). München: Carl Hanser Verlag.

Kippenhahn, R. (2012). *Verschlüsselte Botschaften: Geheimschrift, Enigma und digitale Codes.* (überarbeitete, erweiterte und neuaufgelegte Ausg.). Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag.

Küsters, R., & Wilke, T. (2011). *Moderne Kryptographie: Eine Einführung.* Wiesbaden: Vieweg+Teubner.

Schmeh, K. (2016). *Kryptografie: Verfahren, Protokolle, Infrastrukturen* (6. aktualisierte Ausg.). Heidelberg: dpunkt.Verlag.

Singh, S. (2000). *Geheime Botschaften: Die Kunst der Verschlüsselung von der Antike bis in die Zeiten des Internet.* (K. Fritz, Übers.) München und Wien: Carl Hanser Verlag.

Vinck, H. A. (27. 01. 2014). *Diffie-Hellman-Schlüsseltausch.* Abgerufen am 07. 02. 2024 von Wikipedia. Die freie Enzyklopädie: https://de.wikipedia.org/wiki/Diffie-Hellman-Schl%C3%BCsselaustausch#/media/Datei:Diffie-Hellman_Key_Exchange_(de).svg

Wong, D. (2023). *Kryptografie in der Praxis: Eine Einführung in die bewährten Tools, Frameworks und Protokolle.* (F. Langenau, Übers.) Heidelberg: dpunkt.Verlag.

9. Abbildungsverzeichnis

10. Selbstständigkeitserklärung

Name: Julia Klarissa Grün

Ich erkläre, dass ich diese vorwissenschaftliche Arbeit eigenständig angefertigt und nur die im Literaturverzeichnis angeführten Quellen und Hilfsmittel benutzt habe.

Ort, Datum	Unterschrift (SchülerIn)