Duale Hochschule Baden-Württemberg Mannheim

Portfolio

Diffie-Hellman key exchange

Studiengang Wirtschaftsinformatik

Studienrichtung Data Science

Verfasser:	Franziska Marb, Matthias Fast, Jan Mühlnikel
Matrikelnummern:	5288260, 4750990, 2235021
Kurs:	WWI21DSA
Studiengangsleiter:	Prof. Dr.-Ing. habil. Dennis Pfisterer
Dozent:	Prof. Dr. Maximilian Scherer
Modul:	Integrationsseminar
Bearbeitungszeitraum:	13.11.2023 – 26.02.2024
Eingereicht:	26.02.2024

Disclaimer

Die nachfolgende Arbeit wurde als Gruppenarbeit von allen Verfassern geschrieben und soll daher mit einer Gesamtnote für alle Verfasser bewertet werden.

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Die folgende Ausarbeitung widmet sich dem Diffie-Hellman-Key-Exchange-Algorithmus und wurde zusammen mit einem Erklärungsvideo erstellt und enthält daher einerseits eine detaillierte Betrachtung des Diffie-Hellman-Key-Exchange-Algorithmus und andererseits ein zusätzliches Kapitel, das sich mit der Visualisierung solcher Algorithmen beschäftigt.

1.1 Hintergrundinformationen zur Verschlüsselung

Verschlüsselung gehört zum digitalen Zeitalter und ist heute die Grundlage für einen sicheren Internetverkehr. Heutzutage wird Verschlüsselung als selbstverständlich angesehen und von den Nutzern vorausgesetzt. In den 1970er Jahren schienen die heutigen Verschlüsselungstechnologien noch undenkbar. Dennoch galt es als selbstverständlich, dass dieses Problem nur durch ein so genanntes „geteiltes Geheimnis (engl. shared secret)“ gelöst werden konnte. Diese Idee lässt sich bis ins Römische Reich zurückverfolgen, konnte aber bis heute nicht vollständig gelöst werden [1, S. 268].

Verschlüsselung kann auch unter dem Oberbegriff Kryptographie zusammengefasst werden. Kryptographie ist die Art und Weise, Daten so zu speichern und auszutauschen, dass sie nur von befugten Personen eingesehen werden können. Verschlüsselung ist dabei der Prozess der Umwandlung von Klartext in Chiffretext mit geeigneten Verfahren, Algorithmen und einem Schlüssel, so dass die verschlüsselte Nachricht nur vom vorgesehenen Empfänger mit dem entsprechenden Entschlüsselungsalgorithmus und Schlüssel entschlüsselt werden kann [2, S.28].

1.2 Notwendigkeit von sicheren Schlüsselaustauschprotokollen

In den Anfängen der Computertechnologie wurde dem Datenschutz kaum Beachtung geschenkt. Erst als im Laufe der Jahre Sicherheitsvorfälle auftraten, wurde die Bedeutung des Schutzes digitaler Daten erkannt. Zunächst galten Computerdaten zwar als nützlich,

aber nicht als besonders schützenswert. Diese Sichtweise änderte sich grundlegend, als personenbezogene und finanzielle Daten zunehmend elektronisch verarbeitet wurden und die Relevanz der Datensicherheit in den Vordergrund rückte. Die Gesellschaft erkannte, dass Computer eine immer wichtigere Rolle spielten.

Die ersten Schritte zum Schutz von Daten umfassten Maßnahmen wie Benutzer-IDs und Passwörter, um die Identität der Benutzer zu verifizieren, sowie die Verschlüsselung von Informationen in Datenbanken, um sie vor unbefugtem Zugriff zu schützen. Mit der Weiterentwicklung der Technologie und der zunehmenden Vernetzung wurde jedoch deutlich, dass diese Sicherheitsmaßnahmen nicht mehr ausreichten. Insbesondere mit dem Aufkommen des Internets und der digitalen Kommunikation traten Sicherheitsrisiken auf, die dringend einer Lösung bedurften.

Ein anschauliches Beispiel für Datensicherheitsrisiken ist der Online-Einkauf mit Kreditkarte. Dabei werden sensible Informationen wie Zahlungsdetails und Kreditkarteninformationen über das Internet übertragen und auf den Servern des Händlers gespeichert. Diese Prozesse sind anfällig für Sicherheitslücken, wenn beispielsweise Kreditkarteninformationen während der Übertragung abgefangen oder Datenbanken des Händlers kompromittiert werden. Die daraus resultierenden finanziellen Verluste und Datenschutzverletzungen haben die Notwendigkeit deutlich gemacht, fortschrittliche und sichere Verfahren zur Datenverschlüsselung und -sicherung zu entwickeln [3, S.2 ff.].

1.3 Einführung in den Diffie-Hellman-Key Exchange

Eine Methode, um einen sicheren Datenaustausch zu erreichen, ist der Diffie-Hellman-Key-Exchange. Der Algorithmus, der 1976 von Whitfield Diffie und Martin Hellman 1976 erfunden wurde, ermöglicht es zwei Benutzern, einen Schlüssel sicher auszutauschen, der dann zur Entschlüsselung von Nachrichten verwendet werden kann. Der Algorithmus selbst beschränkt sich auf den Austausch dieser Schlüssel [4, S.1]. Auf die Funktionsweise und die Eigenschaften dieses Algorithmus wird in den folgenden Kapiteln näher eingegangen.

1.4 Geschichte und Bedeutung

Erste Ansätze zur Verschlüsselung lassen sich bis in die Antike zurückverfolgen. Eine der bekanntesten Methoden war damals die Transpositions-Chiffre. Dabei wurde das Alphabet umgestellt oder die Reihenfolge des Alphabets innerhalb eines Wortes verändert.

Klassische Verschlüsselungen boten Schutz und Geheimhaltung, bis im Mittelalter (ca. 800 n. Chr.) die Frequenzanalyse entdeckt wurde. Damit war es möglich, Verschlüsselungen leicht zu brechen. Um 1467 n. Chr. entwickelte der italienische Mathematiker Leon Battista Alberti polyalphabetische Chiffren.

Im 19. Jahrhundert entwickelte Charles Babbage Techniken zur Lösung polyalphabetischer Chiffren. Im Jahr 1917 erfand Gilbert Vernam einen Telegraphenschlüssel, bei dem ein zuvor auf Papierstreifen vorbereiteter Schlüssel Zeichen für Zeichen mit der Klartextnachricht kombiniert wurde, um den Chiffretext zu erzeugen. Dies führte zur Entwicklung von elektromechanischen Chiffriermaschinen und zum One-Time-Pad, einer unknackbaren Verschlüsselung. Zwischen 1920 und 1930 erfand Arthur Scherbius die Rotorchiffriermaschine Enigma, die von der deutschen Armee verwendet wurde.

Nach den beiden Weltkriegen wurde die Aufgabe der Ver- und Entschlüsselung von Maschinen auf Computer übertragen. Die moderne Kryptographie basiert auf Computern und mathematischen Funktionen, um Daten sicherer zu machen. Dazu gehören symmetrische und asymmetrische Verfahren.

Symmetrische Verfahren verwendet denselben Schlüssel für die Ver- und Entschlüsselung. Das bedeutet, dass derselbe Schlüssel sowohl zum Senden als auch zum Empfangen von Daten verwendet wird, oder dass der Verschlüsselungsschlüssel ein anderer ist als der Entschlüsselungsschlüssel, aber beide können durch eine mathematische Funktion miteinander verrechnet werden. Asymmetrische Verfahren verwenden getrennte Schlüssel zum Verschlüsseln und Entschlüsseln von Daten, um die Schwierigkeiten der Schlüsselverwaltung bei symmetrischen Schlüsseln zu umgehen. Bei symmetrischen Schlüsseln muss der Schlüssel zunächst sicher zwischen Sender und Empfänger übertragen werden, was in der realen Welt nicht immer zuverlässig ist.

Um die Nachteile des symmetrischen Verschlüsselungsverfahrens zu überwinden, schlugen Whitfield Diffie und Martin Hellman 1976 den Diffie-Hellman-Key-Exchange-Algorithmus vor, bei dem zwei mathematisch zusammenhängende Schlüssel verwendet werden [2, S.28 ff.].

2 Visualisierung

Dieses Kapitel beschäftigt sich mit den Grundlagen und Möglichkeiten der Visualisierung von Algorithmen in Erklärvideos. Eine genaue Betrachtung ist wichtig, um zu verstehen, wie der Mensch auf bestimmte Animationen und grafische Inhalte reagiert und welche Methoden geeignet sind, um die Aufmerksamkeit des Betrachters zu gewinnen.

2.1 Grundlagen der Visualisierung

Computer-Visualisierungen sind visuelle Umsetzungen von Daten, die den Menschen bei seiner Arbeit unterstützen. Solche Visualisierungen erweitern die menschlichen Fähigkeiten, anstatt sie durch Computer zu ersetzen. Das Feld des Designs ist groß und komplex, da sowohl bei der Erstellung als auch bei der Nutzung viele Aspekte berücksichtigt werden müssen. Gutes Design ist eine Gratwanderung, und für jede Aufgabe muss die Wirksamkeit sorgfältig geprüft werden. Die Grenzen der Computertechnologie, der menschlichen Verarbeitung und der Bildschirme müssen berücksichtigt werden. Visualisierungen werden danach analysiert, warum sie benötigt werden, welche Daten sie darstellen und wie sie aufgebaut sind [5, S.1].

Visualisierung kann in verschiedene Typen unterteilt werden. Wissenschaftliche Visualisierung ist die Verwendung interaktiver visueller Darstellungen von wissenschaftlichen Daten, typischerweise auf physikalischer Basis, um die kognitive Wahrnehmung zu verbessern. Informationsvisualisierung bezieht sich auf die Verwendung interaktiver visueller Darstellungen von abstrakten, nicht physikalisch basierten Daten, ebenfalls mit dem Ziel, die kognitive Wahrnehmung zu verbessern. Während die wissenschaftliche Visualisierung exakte Darstellungen der realen Welt umfasst, befasst sich die Informationsvisualisierung mit der Darstellung oft abstrakter Konzepte. Ziel der Informationsvisualisierung ist es, Datenanalysten dabei zu unterstützen, interne mentale Modelle des Informationsgehalts von Datensätzen zu entwickeln, die das Verständnis erleichtern sollen.

Zudem kann die Visualisierung in drei weitere Kategorien unterteilt werden. Zum einen die explorative Analyse, die der Hypothesenfindung dient. Zum anderen die konfirmatorische Analyse, die die Bestätigung oder Widerlegung von Hypothesen zum Ziel hat. Schließlich

gibt es noch die Präsentation, bei der es um die Darstellung von im Voraus festgelegten Fakten geht [6, S.1 f.].

2.2 Perzeption

2.2.1 Theorie von Malim

Wahrnehmung beschreibt die Art und Weise, wie Menschen etwas wahrnehmen und muss daher grundlegend verstanden werden, wenn visuelle Inhalte produziert werden sollen. Perzeption kann mit der Theorie von Malim aus dem Jahr 1994 beschrieben werden, welche besagt, dass die Wahrnehmung in vier chronologisch aufeinanderfolgenden Phasen abläuft.

Zunächst nimmt der Körper einen Reiz über die Sinnesorgane auf und verarbeitet ihn zu einer ersten Empfindung. Diese Wahrnehmung wird dann weiter analysiert, indem sie mit bereits bekanntem Wissen und dem aktuellen Kontext verglichen wird. Dieser von oben nach unten (top-down) ablaufende Prozess macht uns unsere Wahrnehmung bewusst. Hierbei spielt die kognitive Psychologie eine Rolle. Schließlich wird das Ergebnis dieses Vergleichs interpretiert und erhält eine persönliche Bedeutung, die für die Reaktion des Individuums entscheidend ist. Diese Reaktion ist das Endprodukt des Wahrnehmungsprozesses [7, S.10].

2.2.2 Die Gestalt Prinzipien

Die Gestalt Prinzipien beschreiben die Prinzipien oder Gesetze, nach denen die menschliche Perzeption funktioniert. Sie beschreiben, wie Menschen ähnliche Elemente zu Gruppen zusammenfassen, wie sie Muster erkennen und wie sie komplexe Bilder vereinfachen, wenn sie bestimmte Objekte wahrnehmen. Designer nutzen diese Prinzipien, um Inhalte ästhetisch und leicht verständlich zu präsentieren.

Die grundlegenden Gestalt Prinzipien sind in Abbildung 2.1 dargestellt. Bei der „Proximity“ Regel werden benachbarte Elemente als Gruppe wahrgenommen. Der Benutzer sieht diese Elemente also als Einheit. Die „Similarity“ Regel besagt, dass Elemente, die Ähnlichkeiten aufweisen, als Gruppe wahrgenommen werden. Diese Ähnlichkeit kann zum Beispiel

durch die Form repräsentiert werden. Das „Continuty“ Prinzip besgat, dass Menschen Elemente zusammenfassen, die einem kontinuierlichen Pfad in eine Richtung folgen. Das menschliche Auge folgt diesem Pfad und findet es einfacher, einen kontinuierlichen Weg zu sehen als einzelne Elemente. Die „Closure“ Regel besagt, dass Menschen vollständige Formen bevorzugen, so dass sie automatisch die Lücken zwischen den Formen ausfüllen, um ein vollständiges Bild zu erhalten. Auf diese Weise sieht das menschliche Auge zuerst die vollständige Form, so dass die „Closure“ verwendet werden kann, um das Vertrauen des Benutzers zu erhalten, da er direkt bekannte Formen erkennen kann. Das „Figure-Ground“ Prinzip verdeutlicht, dass Menschen Unsicherheit nicht mögen und daher nach klaren Objekten in einem Bild suchen. Dies bedeutet zum Beispiel, dass Elemente durch Kontrast in den Vordergrund gerückt werden sollten [8].

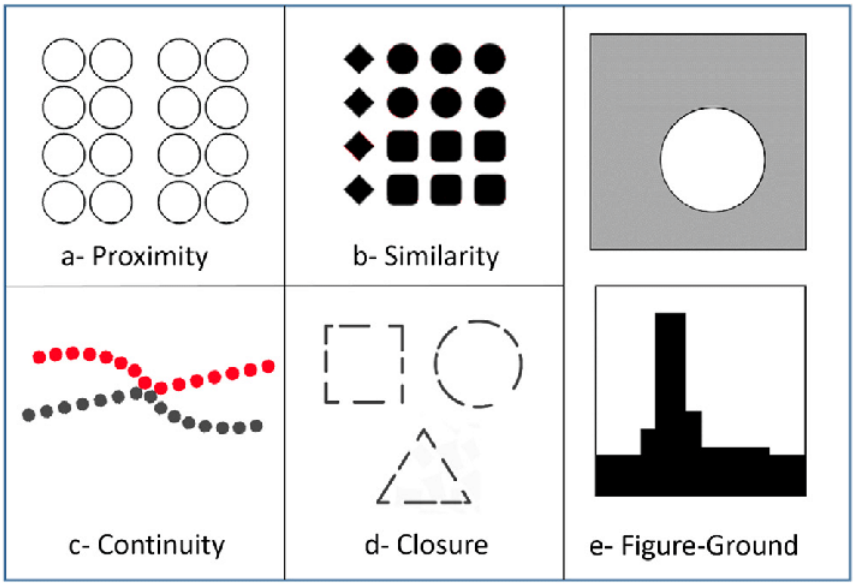

Abbildung 2.1: Grundlegende Gestalt Prinzipien
Quelle: [9]

2.3 Computergrafik

Computergrafik ist definiert als eine Gruppe von Methoden und Techniken zur Umwandlung von Daten in Bilder, die von einem Grafikgerät ausgegeben werden. Die Hauptfrage in diesem Bereich ist, wie Daten in aussagekräftige Bilder umgewandelt werden können [10, S.1].

In der Computergrafik gibt es verschiedene Anwendungsbereiche, die jedoch alle auf ähnliche Schritte zurückgreifen, um Bilder zu erzeugen, die dann auf Computerbildschirmen dargestellt werden. Jeffrey J. McConnell stellt eine Pipeline vor, die diesen Prozess der Erzeugung eines Bildes auf dem Computer darstellt und unterteilt ihn in drei Hauptphasen, wie in Abbildung 2.2 zu sehen ist.

Modellierung

Im ersten Schritt wird ein digitales Modell erstellt. Man entscheidet, wie die Objekte aussehen sollen und wo sie im Raum platziert werden. Es ist ein bisschen so, als würde man eine Szene auf einer Theaterbühne aufbauen, wo jedes Objekt seinen eigenen Platz und seine eigene Form hat.

Rendering

Nachdem das Modell erstellt wurde, wird es im Rendering-Prozess bearbeitet, um es realistisch aussehen zu lassen. Hier werden Lichteffekte, Schatten und Texturen hinzugefügt, um Tiefe und Realismus zu erzeugen. Dieser Schritt sorgt dafür, dass das Modell weniger wie eine Zeichnung und mehr wie ein Foto aus der realen Welt aussieht.

Anzeige

Der letzte Schritt ist die Darstellung. Das gerenderte Bild wird auf einem Gerät wie einem Computerbildschirm oder einem Fernsehgerät angezeigt. Bei Animationen oder Videospielen, bei denen sich die Bilder bewegen, gibt es zusätzliche Schritte, um sicherzustellen, dass die Übergänge zwischen den Bildern fließend sind und die Bewegungen natürlich aussehen [11, S.4 f.].

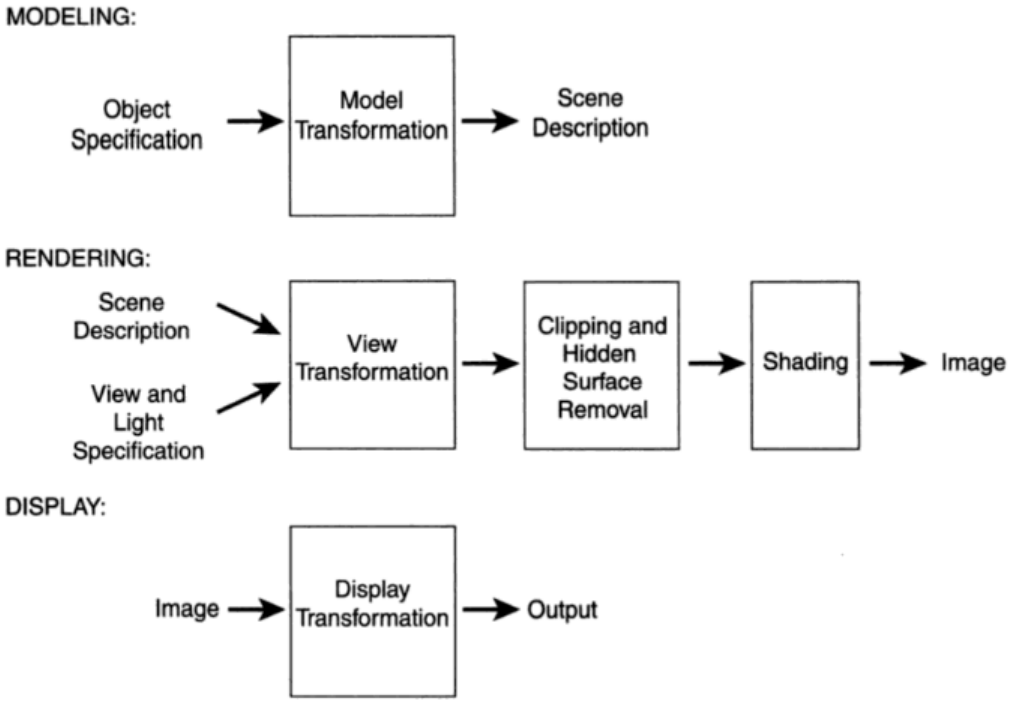

Abbildung 2.2: Computergrafik Pipeline
Quelle: [11, S.5]

2.4 Animation

Im den letzten Jahren hat sich die Aufmerksamkeit auf Akademische Videos und Filme stark erhöht. Animation wird allgemein als eine Abfolge von Bildern verstanden, die sich dynamisch verändern. Das Ziel dabei ist, dem Betrachter zu ermöglichen, eine kontinuierliche Veränderung oder Entwicklung über einen bestimmten Zeitraum hinweg wahrzunehmen. Einfach ausgedrückt, hilft Animation dabei, eine Geschichte oder einen Vorgang durch das Bewegen von Bildern zu erzählen, sodass es aussieht, als würde sich das Gezeigte vor unseren Augen entfalten oder bewegen [12, S.1296].

Im akademischen Umfeld hat sich Animation als Lernmethode längst durchgesetzt. Die Befürworter dieser Lernmethode argumentieren vor allem damit, dass Animationen das Lernen erleichtern, weil sie Bewegung zeigen. Dadurch wird die geistige Anstrengung, sich vorzustellen, wie etwas funktioniert, reduziert. Statt lange darüber nachzudenken, wie ein Prozess abläuft, kann man ihn direkt sehen. Dieser positive Einfluss wird teilweise durch Berichte über den Einsatz von Animationen bei Studierenden der Elektrotechnik

unterstrichen. Dort wurden Animationen im zweiten Studienjahr eingesetzt und es konnte eine Steigerung des Interesses der Studierenden festgestellt werden, jedoch nicht des Verständnisniveaus. Andere Berichte zeigen jedoch auch, dass Animationen zu einer Verbesserung der Leistungen der Studierenden und zu einer positiven Einstellung zum Lernen führen [12, S.1296].

3 Schlüsseltauschproblem

3.1 Definition des Schlüsseltauschproblems

Der Schlüsselaustausch ist wichtig für den sicheren Dateitransfer und ermöglicht es zwei Parteien, Schlüssel über ein unsicheres Netzwerk wie das Internet auszutauschen. Dieser Prozess verwendet hauptsächlich Algorithmen wie RSA oder den hier betrachteten Diffie-Hellman-Algorithmus.

Um die Vertraulichkeit der Daten während der Übertragung zu schützen, verwenden Protokolle wie FTPS, HTTPS und SFTP symmetrische Verschlüsselung, die einen gemeinsamen Schlüssel für beide Kommunikationspartner erfordert. Die Herausforderung besteht darin, diesen gemeinsamen Schlüssel sicher zu übertragen, insbesondere wenn die Partner weit voneinander entfernt sind oder sich noch nie begegnet sind. Da die direkte Übertragung des Schlüssels über konventionelle Wege Sicherheitsrisiken birgt, wurden Schlüsselaustauschprotokolle entwickelt. Diese ermöglichen den Austausch symmetrischer Schlüssel über unsichere Netzwerke wie das Internet, sind einfach zu handhaben, sicher, skalierbar und für den schnellen, unsicheren Datenaustausch konzipiert. Sie sind ideal für den Einsatz bei sensiblen und weitreichenden Transaktionen [13].

3.2 Aspekte des Schlüsselaustausches

Das Problem des Schlüsselaustausches umfasst im Allgemeinen zwei Aspekte. Einerseits wird durch die Authentisierung die Identität der beteiligten Parteien sichergestellt, andererseits wird durch die Geheimhaltung gewährleistet, dass der Schlüssel keinem Gegner oder Eindringling bekannt ist.

Diese beiden Aspekte können in unterschiedlichen Formen auftreten. Die Authentifizierung kann in verschiedenen Formen erfolgen, entweder einseitig, wobei nur eine Partei die Identität der anderen Partei überprüft, oder gegenseitig, wobei sich beide Parteien identifizieren. Sie kann sich direkt auf die am Austausch beteiligten Akteure beziehen, bekannt als Entitätsauthentifizierung, oder indirekt durch die Überprüfung der Identitäten, die

zu den verwendeten Schlüsseln gehören, bekannt als Schlüsselauthentifizierung. Die Authentifizierung kann explizit während des Austauschs erfolgen oder implizit, abhängig von der Verwendung des Sitzungsschlüssels in anderen Protokollen. Häufig wird auch eine Schlüsselbestätigung erwartet, die sicherstellt, dass ein abgeleiteter Schlüssel tatsächlich von der anderen Partei in der Kommunikation verwendet wird, wodurch gewährleistet wird, dass nur korrekt ausgetauschte Schlüssel verwendet werden. Unterschiedliche Formen bilden daher auch unterschiedliche Stärken der Sicherheit ab [14, Section I].

3.3 Problem bei traditionellen Schlüsselaustauschmethoden

Das Problem herkömmlicher Schlüsselaustauschverfahren besteht darin, dass derselbe Schlüssel zum Ver- und Entschlüsseln von Nachrichten verwendet wird [15]. Der Schlüssel spielt bei der traditionellen symmetrischen Verschlüsselung eine sehr wichtige Rolle, da ihre Sicherheit direkt von der Art des Schlüssels, d.h. der Schlüssellänge, abhängt. Die Schwachstelle traditioneller symmetrischer Schlüsselaustauschverfahren lag daher in der gemeinsamen Nutzung symmetrischer Schlüssel, die zwischen Sender und Empfänger ausgetauscht werden mussten und somit leicht abgefangen werden konnten [16, S.878 ff.].

4 Mathematische Grundlagen

Für die Sicherheit moderner kryptografischer Systeme ist es entscheidend, dass das Faktorisieren großer natürlicher Zahlen als schwierig anzusehen ist. Im nachfolgenden Kapitel werden die verschiedenen mathematischen Grundlagen für den Diffie-Hellman-Key-Exchange-Algorithmus erläutert. In diesem Text werden die grundlegenden Operationen der Kryptologie erläutert, wie beispielsweise die Division mit Rest für ganze Zahlen und Polynome oder die Bestimmung von Inversen von Elementen aus der primen Restklassengruppe:

$$Z_n^x = \{a \in Z_n \mid ggT(a, n) = 1\}$$

Einwegfunktionen sind Funktionen, die leicht berechnet werden können, aber nur schwer invertiert werden können. Sie bilden die Basis für alle asymmetrischen kryptografischen Systeme [17, S.57]. Im Anschluss wird das diskrete Logarithmusproblem genauer beschrieben.

4.1 Komplexität und Einwegfunktionen

4.1.1 Division mit Rest

Die *Division mit Rest* für ganze Zahlen, die für die Kryptologie von grundlegender Bedeutung ist, ähnelt der Division mit Rest für Polynome. Sie ist wie folgt definiert [18, S.62]:

Satz 4.1 (Division mit Rest):
Zu ganzen Zahlen a, b, wobei $b \neq 0$, existieren eindeutig bestimmte $q, r \in Z$ mit

$$a = bq + r \quad und \quad 0 \leq r < |b|.$$

Die Notation aus der linearen Algebra sagt aus, dass zwei ganze Zahlen a und r **kongruent** modulo b sind, wenn es einen q in Z mit $a = bq + r$ gibt. Beschrieben auch

$$a \equiv r \pmod{b}.$$

Der Divisionssatz mit Rest besagt, dass im Falle $b \neq 0$ jede ganze Zahl a zu einer Zahl $r \in \{0, \ldots, |b| - 1\}$ modulokongruent ist. Offensichtlich ist $\equiv (mod b)$ eine Äquivalenzrelation, wobei die Anzahl der Äquivalenzklassen durch $|b|$ gegeben ist.

Der Algorithmus für die Division mit Rest basiert auf der folgenden Laufzeit:

Satz 4.2:
Die Division mit Rest bei ganzen Zahlen a, b, wobei $b \neq 0$, d.h. die Bestimmung von $q, r \in Z$ mit $a = bq + r$ und $0 \leq r < |b|$, hat die Laufzeit

$$O(\log b \log q) = O\left(\log b \log \frac{a}{b}\right).$$

Die Multiplikation zweier Zahlen a und b modulo m mit Faktoren $0 \leq a, b < m$ ist eine typische Anwendung: Um den kleinsten positiven Vertreter von $ab \in Z_m$ zu bestimmen, wird die Division mit Rest auf ab angewendet. Es ist zu beachten: a, b und ab sind ungefähr gleich groß wie m. Beide Teilschritte des Algorithmus, d.h. die Multiplikation und die Division mit Rest, sind quadratisch, so dass eine Zusammenfassung möglich ist [17, S.62]:

Satz 4.3:
Die Multiplikation modulo m hat die Laufzeit $O((\log m)^2)$, ist also quadratisch.

Satz 4.4:
Es seien a, b Polynome über einem Körper K, und es gelte $b \neq 0$. Die Division mit Rest, also die Bestimmung von $q, r \in K[X]$ mit $a = bq + r$ und $\deg r < \deg b$, hat die Laufzeit $O(\deg b \cdot \deg q)$.

4.1.2 Euklidischer Algorithmus

Satz 4.5 (Der euklidische Algorithmus):
Der euklidische Algorithmus bricht nach $n + 1$ Schritten ab. Sind a, b ganze Zahlen, $b \neq 0$, so ist der Rest r_n der ggT von a und b [17, S.67], [19, S. 193]:

$$r_n = ggT(a, b).$$

Weiter liefert der Algorithmus ganze Zahlen $x := x_n$ und $y := y_n$ mit

$$ggT(a, b) = ax + by.$$

4.1.3 Prime Restklassengruppen

In diesem Abschnitt ist ein natürliche Zahl $n > 1$ gewählt.

Die Einheit des Restklassenringes $(Z_n, +, \cdot)$ ist die Multiplikationsgruppe $(Z_n^*, \cdot)$, mit

$$Z_n^* = \{a \in Z_n \mid a \text{ ist invertierbar}\} = \{a \in Z_n \mid b \in Z_n \text{ mit } ab \equiv 1 \pmod{n}\},$$

wird auch als **prime Restklassengruppe** modulo n bezeichnet [20, S. 181-183].

Satz 4.6:
Für $a \in Z$ gilt: $a \in Z_n^* \iff ggT(a, n) = 1$.

Beweis: Es sei $a = a + nZ$ invertierbar. Folglich existiert ein $b \in Z$ mit

$$1 + nZ = (a + nZ) \cdot (b + nZ) = ab + nZ.$$

Es gibt also ein $y \in Z$ mit $ab - 1 = ny \in nZ$. Jeder gemeinsame Teiler von a und n ist ein Teiler von 1. Daraus folgt $ggT(a, n) = 1$.

Wenn $ggT(a, n) = 1$, dann existieren nach Satz 4.5 des euklidischen Algorithmus ganze Zahlen b und y mit $ab + ny = 1$, und es gilt

$$(a + nZ) \cdot (b + nZ) = ab + nZ = 1 + nZ.$$

Somit ist $a = a + nZ \in Z_n$ invertierbar.

Nach diesem Satz gilt

$$Z_n^* = \{a \in Z_n \mid ggT(a, n) = 1\}.$$

Beispiel:
Gegeben sind die primen Restklassengruppen für einige kleine n:
$Z_2^x = \{1\}, Z_3^x = \{1, 2\}, Z_4^x = \{1, 3\}, Z_5^x = \{1, 2, 3, 4\},$
$Z_6^x = \{1, 5\}, Z_7^x = \{1, 2, 3, 4, 5, 6\}, Z_8^x = \{1, 3, 5, 7\}.$
In Z_8^x ist jedes Element zu sich selbst invers, und es gilt etwa $3 \cdot 5 = 7$.

Wie der Beweis von Satz 4.6 zeigt, ist es mit dem erweiterten euklidischen Algorithmus nicht nur möglich zu prüfen, ob $a \in Z$ modulo n invertierbar ist, sondern gegebenenfalls auch das Inverse zu bestimmen. Da dies in der Kryptologie von grundlegender Bedeutung ist, wird darauf explizit hingewiesen.

Satz 4.7:
Es sei $a \in Z, -n < a < n$, mit $ggT(a,n) = 1$ gegeben. Der erweiterte euklidische Algorithmus 4.10 liefert $b, y \in Z$ mit $ab + ny = 1$ in der Laufzeit $O((\log n)^2)$. Es ist b das Inverse von a im Ring Z_n.

Beispiel: Nachfolgend wird mit dem erweiterten euklidischen Algorithmus das Inverse von 351 in Z_{770} bestimmt:

Schritt	Division	Darstellung	r_i	x_i	y_i
1	$770 = 2 \cdot 351 + 68$	$r_1 = 770x_1 + 351y_1$	68	1	-2
2	$351 = 5 \cdot 68 + 11$	$r_2 = 770x_2 + 351y_2$	11	-5	11
3	$68 = 6 \cdot 11 + 2$	$r_3 = 770x_3 + 351y_3$	2	31	-68
4	$11 = 5 \cdot 2 + 1$	$r_4 = 770x_4 + 351y_4$	1	-160	351
5	$2 = 2 \cdot 1 + 0$	$r_5 = 0$	Stop!		

Erhalten wird

$$1 = 770 \cdot (-160) + 351 \cdot 351, d.h. 1 = 351 \cdot 351,$$

somit ist 351 das Inverse zu 351 in Z_{770} [21, S. 202], [17].

4.1.4 Einwegfunktion

Eine Funktion f, für die $b = f(a)$ leicht zu berechnen ist, für die aber $a \in f^{-1}(\{b\})$ schwer zu bestimmen ist, wird als Einwegfunktion bezeichnet. Eng verbunden mit der Frage, ob die Komplexitätsklassen P und NP verschieden sind, ist die Existenz echter Einwegfunktionen. Da diese Frage offen ist, können über die Existenz von Einwegfunktionen nur Vermutungen angestellt werden.

Im Folgenden wird gelegentlich von einer Funktion in P oder NP gesprochen. Dies ist ein Hinweis auf die Existenz eines Algorithmus zur Auswertung von f, der in der entsprechenden Klasse ist. In vielen Fällen, die für diesen Abschnitt von Bedeutung sind, ist dies von der Art der Beschreibung von f abhängig [22, S.254ff.], [23, S.132ff.], [17, S.76-80].

Seien A und B Mengen. Eine Abbildung $f : A \to B$ wird als Einwegfunktion bezeichnet, wenn folgendes gilt:

(E1) $f \in P$ und f lässt sich tatsächlich effizient berechnen;

(E2) Das Problem Π „zu $b \in B$ bestimme $a \in A$ mit $f(a) = b$“, d.h., finde ein Element in $f^{-1}(b)$, ist für die meisten $b \in B$ nicht effizient lösbar. Im besten Fall ist Π nicht in P.

Es ist zu beachten, dass das Problem Π in NP liegt. Dies ist auf die erste Bedingung zurückzuführen.

Beispiel:
Es wird die Menge $I_k = [2^{k-1}, 2^k - 1] \cap N$ der k-Bit-Zahlen betrachtet. Die Multiplikationstabelle

$$I_k \times I_k \to N, \quad (a, b) \mapsto ab$$

kann auf effiziente Weise ausgeführt werden. Daher ist (E1) erfüllt. Allerdings ist (E2) nicht erfüllt, da die Formulierung für die meisten Anwendungen nicht geeignet ist. Für kryptographische Anwendungen werden die Ergebnisse zu oft faktorisiert.

Wird diese Abbildung jedoch eingeschränkt, indem nur k-Bit-Primzahlen, d.h. Elemente der Menge $P_k := \{p \in I_k; pprim\}$, zugelassen werden, so erhält man

$$P_k \times P_k \to N, \quad (a, b) \mapsto ab.$$

Nach allem, was wir heute wissen, handelt es sich bei dieser Abbildung vermutlich um eine Einwegfunktion. Diese Einwegfunktion ist die Grundlage des bekannten RSA-Verfahrens oder auch des Diffie-Hellman-Verfahrens [22, S.254ff.], [23, S.132ff.], [17, S.76-80].

4.2 Diskrete Logarithmusproblem

Auf der Schwierigkeit, diskrete Logarithmen zu bestimmen, beruht die Sicherheit vieler kryptographischer Verfahren. Beispiele hierfür sind das Diffie-Hellman-Verfahren zum Austausch von Schlüsseln und das ElGamal-Verfahren, auch in seiner Variante als Signaturverfahren. Für die Kryptologie ist das Problem der diskreten Logarithmen daher von großer Bedeutung.

Sei p eine Primzahl. g sei ein erzeugendes Element von Z und x sei eine ganze Zahl. Die Funktion $exp : Z \to Z$, $x \mapsto g^x p$ wird diskrete Exponentialfunktion genannt. Die Umkehrfunktion der diskreten Exponentialfunktion wird als diskrete logarithmische Funktion bezeichnet. Ist g ein Erzeugungselement von Z, dann existiert zu jedem Element von Z der diskrete Logarithmus, da jede Zahl von Z eine Potenz von g ist. Der diskrete Logarithmus von y zur Basis g ist die kleinste natürliche Zahl x mit $y = g^x p$.

Wenn g kein erzeugendes Element ist, gibt es nicht notwendigerweise einen diskreten Logarithmus für jedes $y\ inZ$. Ein erzeugendes Element g von Z wird im Folgenden als gegeben angenommen.

Ein effizienter Algorithmus zur Bestimmung des diskreten Logarithmus ist bis heute nicht bekannt. Dass sich die diskrete Exponentialfunktion chaotischer verhält als ihre stetige Entsprechung, zeigt Abbildung 4.1.

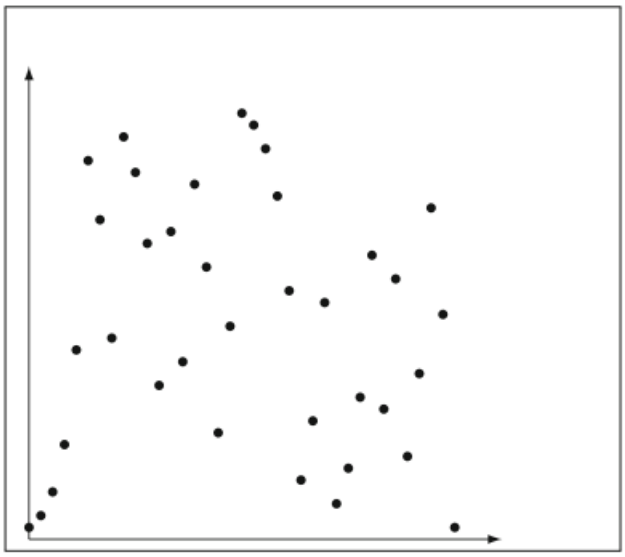

Abbildung 4.1: Der diskrete Charme der Exponentialfunktion
Quelle:

Ein naiver Ansatz für die Bestimmung der diskreten Logarithmen besteht darin, alle Zahlen auszuprobieren, die für die Bestimmung in Frage kommen. Da g ein erzeugendes Element ist, ist die Ordnung von g die Zahl $p-1$, was insbesondere bedeutet, dass $g^{p-1}p = g^0 = 1$ ist, und der diskrete Logarithmus einer beliebigen Zahl $y \in Z$ muss nur in der Menge $\{0, \ldots, p-2\}$ gesucht werden. Eine solche vollständige Suche ist praktisch undurchführbar, wenn p eine sehr große Primzahl ist.

Der Baby-Step-Giant-Step-Algorithmus ist ein weniger naiver Ansatz zur Berechnung des diskreten Logarithmus und deutlich effizienter als die vollständige Suche. Allerdings ist auch dieser Algorithmus nicht als effizient zu bezeichnen. So wird keine polynomiale Laufzeit erreicht [24, S.179ff.], [23, S.135], [22, S.256].

5 Funktionsweise des Diffie-Hellman-Key Exchange

Im Allgemeinen sind symmetrische Verschlüsselungsverfahren effizienter als Public-Key-Verfahren. Eine Möglichkeit, dies zu nutzen und trotzdem das Problem der Schlüsselverwaltung zu lösen zu lösen, ist bereits unter dem Stichwort Hybridverfahren bekannt: Verschlüsselung einer Nachricht mit einem (effizienten) symmetrischen Verfahren und des (kurzen) Schlüssels zur Entschlüsselung dieser Nachricht mit einem Public-Key-Verfahren und anschließender Versand beider Teile an den Empfänger. In diesem Kapitel wird eine trickreiche Variante diskutiert, bei der ein Schlüssel über einen unsicheren Kanal ausgetauscht wird, um dann die Kommunikation mit einem symmetrischen Verfahren abzusichern. Beim Diffie-Hellman-Schlüsselaustausch wird ein geheimer Schlüssel über eine öffentliche, ungesicherte Leitung ausgetauscht. Obwohl ein Angreifer beobachten kann, wie Teile des geheimen Schlüssels ausgetauscht werden, gelingt es ihm in der Regel nicht, den Schlüssel selbst aus diesen Teilen zu generieren. Die Schwierigkeit des diskreten Logarithmusproblems liegt in der Sicherheit des Verfahrens [25, S.167], [21, S.187], [26].

Sei p eine Primzahl. In der multiplikativen zyklischen Gruppe $Z_p^\times$ der Ordnung $p-1$ wird der Diffie-Hellman-Schlüsselaustausch beschrieben. Jedes Erzeugungselement der Gruppe $Z_p^\times$ wird als primitive Wurzel modulo p bezeichnet. Laut Korollar hat $Z_p^\times$ genau $\phi(p-1)$ Primitivwurzeln modulo p. ϕ bezeichnet die Euler'sche ϕ-Funktion.

Beispiel:
Es ist 2 eine Primitivwurzel modulo 13, d.h. $2^{o(2)} \equiv 1 \pmod{13}$, da $o(2) = 12$:

k	1	2	3	4	5	6	7	8	9	10	11	12
2^k	2	4	8	3	6	12	11	9	5	10	7	1

Wegen Korollar sind die $\phi(12) = 4$ Primitivwurzeln modulo 13 die vier Elemente $2^1 = 2$, $2^5 = 6$, $2^7 = 11$, $2^{11} = 7$ [23, S.133ff.], [25, S.167f.].

5.1 Schlüsselaustausch

Das Diffie-Hellman-Schlüsselaustauschverfahren ist ein Verfahren zum Austausch eines (geheimen) Schlüssels über einen öffentlichen, ungesicherten Kanal und zur anschließenden Kommunikation mit einem symmetrischen Verfahren.

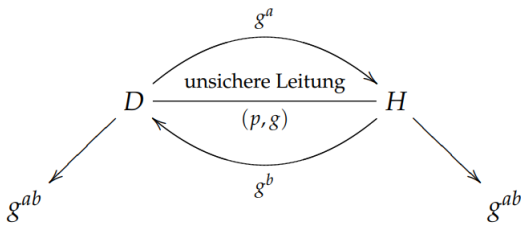

Abbildung 5.1: Diffie-Hellman Schlüsseltausch
Quelle: [25]

Im hiesigen Kontext funktioniert das Verfahren so:

1. D und H einigen sich auf eine Primzahl p und eine Primitivwurzel g modulo p: Es ist (p, g) öffentlich bekannt.
2. D wählt ein $a \in \{2, \ldots, p-2\}$, bestimmt $ga \in Z_p^\times$ und sendet $A := ga$ an H (der Exponent a bleibt geheim).
3. H wählt ein $b \in \{2, \ldots, p-2\}$, bestimmt $gb \in Z_p^\times$ und sendet $B := gb$ an D (der Exponent b bleibt geheim).
4. D berechnet $Ba = gab$, H berechnet $Ab = gab$.

Obwohl gab selbst nicht über den unsicheren Kanal ausgetauscht wurde, besitzen D und H beide den geheimen Schlüssel gab. Entscheidend ist die einfache Tatsache, dass $gab = gba$. Der geheime Schlüssel kann nun z.B. für die Kommunikation mit einem symmetrischen Verfahren verwendet werden.

Beispiel:
Es sei $p = 17$. Es ist 3 eine Primitivwurzel modulo 17. D wählt $a = 7 \to 3^7 = 11$. H wählt $b = 4 \to 3^4 = 13$.

Austausch:

$$3^7 \equiv 11 \pmod{17}$$

$$3^4 \equiv 13 \pmod{17}$$

Es ergibt sich:

$$3^{7 \cdot 4} \equiv 3^{28} \equiv 4 \pmod{17}.$$

D und H haben beide den geheimen Schlüssel 4.

Bemerkung:
In diesem Beispiel mit $g = 3 \in Z_{17}^{\times}$ kann ein Angreifer natürlich sofort g^{ab} aus ga und gb bestimmen. Da $7 = \log_3(11)$ und $4 = \log_3(13)$ leicht zu ermitteln sind, ist $g^{28} = 4$ der geheime Schlüssel. In der Praxis sollte p so gewählt werden, dass das diskrete Logarithmenproblem (siehe Abschnitt 4.2) schwer zu lösen ist [25, S.168f.], [23, S.187ff.], [26].

5.2 Das Diffie-Hellman-Problem

Sei p eine Primzahl. In der multiplikativen zyklischen Gruppe $Z_p^{\times}$ der Ordnung $p - 1$ wird der Diffie-Hellman-Schlüsselaustausch beschrieben.

Ein Angreifer, der den Schlüsselaustausch beobachtet, kennt die Größen

$$p,\ g,\ g^a,\ g^b, \text{ aber nicht } a,\ b,\ g^{ab}.$$

Das Diffie-Hellman-Problem lautet wie folgt:

$$\text{Berechne } g^{ab} \text{ aus den Größen } g,\ g^a \text{ und } g^b.$$

Wenn das diskrete Logarithmusproblem lösbar ist, kann auch das Diffie-Hellman-Problem gelöst werden. Aus den Gleichungen $c = g^a$ und $d = g^b$ werden die diskreten Logarithmen a und b bestimmt und dann g^{ab} mit den dann bekannten Größen g, a, b. Ob es möglich ist, das Diffie-Hellman-Problem zu lösen, ohne die diskreten Logarithmen berechnen zu können, ist bisher nicht bekannt [25, S.169], [23, S.186f.], [26].

5.3 Der Mann in der Mitte

Für das gerade beschriebene Verfahren von Diffie und Hellman - der Mann in der Mitte - wird ein möglicher Angriff beschrieben. Die Größen p und g wurden von den Teilnehmern D und H über einen unsicheren Kanal vereinbart. Ein Angreifer M hat somit die Möglichkeit der Beobachtung des Paares (p, g). Bevor D und H weitere Größen austauschen, stellt sich M zwischen die ahnungslosen D und H und geht dann wie folgt vor:

- M fängt g^a von D ab und leitet ein $g^{a'}$ mit einem nur ihm bekannten a' an H weiter (H denkt, $g^{a'}$ kommt von D).
- M fängt g^b von H ab und leitet ein $g^{b'}$ mit einem nur ihm bekannten b' an D weiter (D denkt, $g^{b'}$ kommt von H).
- D bildet mit seinem a den vermeintlich geheimen Schlüssel $g^{b'a}$.
- H bildet mit seinem b den vermeintlich geheimen Schlüssel $g^{a'b}$.
- Weil M sowohl g^a als auch g^b als auch die Größen a', b' kennt, kann M ebenfalls die Schlüssel $g^{b'a}$ und $g^{ab'}$ bilden.
- M kann und muss jeden Geheimtext von D mit $g^{b'a}$ entschlüsseln und mit $g^{ab'}$ wieder verschlüsselt an H weiterreichen. Analog verfährt er mit Nachrichten von H an D.

Wenn dem Mann M in der Mitte ein Text entgeht, so fliegt er wahrscheinlich auf.

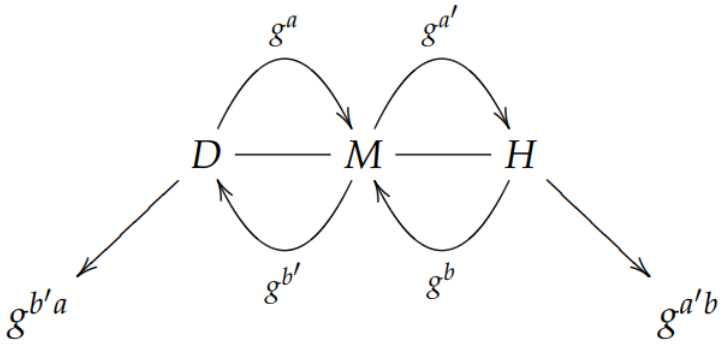

Abbildung 5.2: Mann in der Mitte
Quelle: [25]

Bemerkung

Für alle Verfahren der asymmetrischen Kryptographie stellen Varianten dieses Angriffs eine Bedrohung dar. Die Schwachstelle liegt in der Übertragung des öffentlichen Schlüssels. Wenn es jemandem gelingt, sich an dieser Stelle einzuschalten, ist die gesamte Kommunikation nicht mehr sicher. Eine mathematische Lösung für dieses Problem ist bisher nicht bekannt (vielleicht auch gar nicht möglich). Es ist nur durch besondere Sorgfalt der Benutzerinnen und Benutzer in den Griff zu bekommen [25, S.170f.], [26].

6 Sicherheit im Diffie-Hellman-Verfahren

6.1 Bedrohungen und Angriffsszenarien

Angriffe auf das Diffie-Hellman-Verfahren können auf verschiedene Weisen geschehen:

6.1.1 Denial of service Attacks

Bei Denial-of-Service-Angriffen versucht der Angreifer, die Teilnehmer daran zu hindern, ihre Kommunikationsprotokolle erfolgreich auszuführen. Bei diesen Angriffen versucht der Angreifer, den normalen Betrieb eines Dienstes, Systems oder Netzes zu stören oder ganz zu unterbrechen. Einer der in der Praxis wirksamsten Angriffe besteht darin, Server mit Anfragen zu überlasten. Dabei handelt es sich um eine Form des Denial-of-Service-Angriffs, da der Server mit der Bearbeitung der gefälschten Anfragen so beschäftigt ist, dass er keine Zeit mehr hat, auf legitime Anfragen zu antworten.

6.1.2 Outsider Attacks

Der Angreifer versucht, das Protokoll zu stören, indem er beispielsweise Nachrichten hinzufügt, entfernt oder wiederholt. Ziel ist es, an interessante Informationen zu gelangen, also Informationen, die er nicht erhalten könnte, wenn er sich einfach die öffentlichen Schlüssel ansieht.

6.1.3 Insider Attacks

Ein Teilnehmer des Diffie-Hellman-Protokolls könnte absichtlich einen Protokolldurchlauf so gestalten, dass er angreifbar ist, so dass ein Außenstehender das gemeinsam vereinbarte Geheimnis herausfinden kann. Wenn sich ein Teilnehmer entscheidet, das gemeinsame Geheimnis preiszugeben, kann dagegen natürlich nichts unternommen werden. Au-

ßerdem ist zu bedenken, dass Schadsoftware bei der Durchführung eines solchen Angriffs sehr effektiv sein kann.

6.1.4 Man in the Middle Attacks

Ein aktiver Angreifer, der in der Lage ist, Nachrichten zu entfernen und hinzuzufügen, kann ein Kommunikationsprotokoll leicht kompromittieren. Durch das Abfangen und Ersetzen von Nachrichten zwischen den Kommunikationspartnern kann der Angreifer beide Seiten glauben machen, dass sie einen gemeinsamen geheimen Schlüssel teilen. In Wirklichkeit glaubt jeder der Kommunikationspartner, dass der Schlüssel etwas ist, das der Angreifer bestimmt hat. In einem Szenario, in dem dieser geheime Schlüssel zur symmetrischen Verschlüsselung von Nachrichten verwendet wird, könnte der Angreifer die verschlüsselte Kommunikation manipulieren. Er könnte Nachrichten abfangen, entschlüsseln, verändern oder die Übertragung ganz blockieren, ohne dass die Kommunikationspartner dies bemerken.

6.1.5 Weitere Angriffe

Andere Angriffe können zum Beispiel auf der Zahlentheorie basieren. Zu diesen Angriffen gehören auch die „Degenerate Message Attacks“, die zum Beispiel einfache Exponenten ausnutzen. Es gibt weitere Algorithmen, die mit Hilfe der Zahlentheorie das Diffie-Hellman-Verfahren brechen. Leider würde eine genauere Erläuterung dieser Algorithmen den Rahmen dieser Arbeit sprengen. Erwähnenswert sind zudem Angriffe bezogen auf die Authentifizierung und auf Implementierungsdetails [27, S.6 ff.].

6.2 Stärkung der Sicherheit

Nils Mäurer et al. stellen in ihrer Arbeit drei Varianten des Diffie-Hellman-Key-Exchange vor um diesen sicherer zu machen:

6.2.1 Ephemeral Diffie-Hellman-Key Exchange

Dies ist die traditionelle Methode, bei der große Schlüssel verwendet werden, um die Sicherheit zu gewährleisten. Um die Sicherheit gegen Angriffe zu erhöhen, wird sie oft mit zusätzlichen Sicherheitsmaßnahmen wie dem STS-Protokoll kombiniert. Die empfohlene Schlüsselgröße beträgt mindestens 3072 Bit.

6.2.2 Elliptic Curve Diffie-Hellman-Key Exchange (ECDH)

Diese Methode verwendet elliptische Kurven, um mit viel kleineren Schlüsselgrößen, zum Beispiel 256 Bit, ein ähnliches Sicherheitsniveau wie bei der traditionellen DHKE zu erreichen. Dies macht sie effizienter und schneller, ohne die Sicherheit zu beeinträchtigen.

6.2.3 Supersingular Isogeny Diffie-Hellman-Key Exchange (SIDH)

Diese fortgeschrittene Variante ist resistent gegen künftige Angriffe von Quantencomputern. Sie verwendet komplexe mathematische Strukturen für den Schlüsselaustausch und kann sichere Verbindungen auch mit kleineren Schlüssellängen von 2640 Bit herstellen [28, Section II].

Literaturverzeichnis

[1] J. H. Ellis, „THE HISTORY OF NON-SECRET ENCRYPTION," *CRYPTOLOGIA*, Jg. 23, Nr. 3, S. 267–273, 1999. DOI: 10.1080/0161-119991887919.

[2] D. Pandya, K. R. Narayan, S. Thakkar, T. Madhekar und B. Thakare, „Brief history of encryption," *International Journal of Computer Applications*, Jg. 131, Nr. 9, S. 28–31, 2015.

[3] S. R. Chauhan und S. Jangra, „Computer Security and Encryption: An Introduction," in *Computer Security and Encryption*, S. R. Chauhan und S. Jangra, Hrsg., Boston, MA und Paris: Mercury Learning and Information und Cyberlibris, 2020, ISBN: 9781683925293. Adresse: https://www.degruyter.com/document/doi/10.1515/9781683925293/html.

[4] D.-H. K. Exchange, „Diffie-Hellman Key Exchange," *Diffie% E2*, Jg. 80,

[5] T. Munzner, *Visualization analysis and design*. CRC press, 2014.

[6] H. Nagel, „Scientific Visualization versus Information Visualization," Jan. 2006.

[7] M. Martsch, O. Wienert, S. Liefold und K. Jenewein, „Perzeption in virtueller Realität als Aggregat von Visualisierung und Interaktion," *BBP-Arbeitsberichte*, Jg. 77, 2010.

[8] Interaction Design Foundation - IxDF, *What are the Gestalt Principles?* https://www.interaction-design.org/literature/topics/gestalt-principles, Accessed: 25-Feb-2024, 2016.

[9] Unknown author, *The Potency of Architectural Probabilism in Shaping Cognitive Environments: A Psychophysical Approach*, Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Gestalt-Basic-Principles-34-12_fig2_353105380, Accessed 25 Feb, 2024.

[10] J. Gomes, L. Velho und M. C. Sousa, *Computer graphics: theory and practice*. CRC Press, 2012.

[11] J. J. McConnell, *Computer graphics: theory into practice*. Jones & Bartlett Learning, 2005.

[12] A. Gero und W. Zoabi, „Computer animation and academic achievements: Longitudinal study in electronics education," *International Journal of Engineering Education*, Jg. 30, Nr. 5, S. 1295–1302, 2014.

[13] J. C. Villanueva, *What Is A Key Exchange?* Last Updated: February 22, 2024, Feb. 2024. Adresse: https://www.jscape.com/blog/key-exchange.

[14] C. D. de Saint Guilhem, M. Fischlin und B. Warinschi, „Authentication in key-exchange: Definitions, relations and composition," in *2020 IEEE 33rd Computer Security Foundations Symposium (CSF)*, IEEE, 2020, S. 288–303.

[15] M. Tyson, „Understand Diffie-Hellman Key Exchange," *InfoWorld*, Jan. 2022, Zugriff am 25. Februar 2024. Adresse: https://www.infoworld.com/article/3647751/understand-diffie-hellman-key-exchange.html.

[16] M. Agrawal und P. Mishra, „A comparative survey on symmetric key encryption techniques," *International Journal on Computer Science and Engineering*, Jg. 4, Nr. 5, S. 877, 2012.

[17] C. Karpfinger und H. Kiechle, „Komplexität und Einwegfunktionen," in *Kryptologie: Algebraische Methoden und Algorithmen*. Wiesbaden: Vieweg+Teubner, 2010, S. 57–80, ISBN: 978-3-8348-9356-7. DOI: 10.1007/978-3-8348-9356-7_4. Adresse: https://doi.org/10.1007/978-3-8348-9356-7_4.

[18] P. Hartmann, *Mathematik für Informatiker, Ein praxisbezogenes Lehrbuch* (SpringerLink Bücher), ger, 4., überarbeitete Auflage. Wiesbaden: Vieweg, 2006, ISBN: 9783834891068. DOI: 10.1007/978-3-8348-9106-8.

[19] B. Lenze, „Grundlagen der Zahlentheorie," in *Basiswissen Angewandte Mathematik – Numerik, Grafik, Kryptik: Eine Einführung mit Aufgaben, Lösungen, Selbsttests und interaktivem Online-Tool*. Wiesbaden: Springer Fachmedien Wiesbaden, 2020, S. 187–225, ISBN: 978-3-658-30028-9. DOI: 10.1007/978-3-658-30028-9_12. Adresse: https://doi.org/10.1007/978-3-658-30028-9_12.

[20] K. Schmeh, *Kryptografie, Verfahren, Protokolle, Infrastrukturen* (iX Edition), ger, 6th ed. Heidelberg: dpunkt.verlag, 2016, 1946 S., Schmeh, Klaus (VerfasserIn), ISBN: 9783864919084. Adresse: https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=4688569.

[21] K. Freiermuth, J. Hromkovič, L. Keller und B. Steffen, „Der geheime Schlüsselaustausch und das Diffie-Hellman-Protokoll," in *Einführung in die Kryptologie: Lehrbuch für Unterricht und Selbststudium*. Wiesbaden: Springer Fachmedien Wiesbaden, 2014, S. 185–206, ISBN: 978-3-8348-2269-7. DOI: 10.1007/978-3-8348-2269-7_7. Adresse: https://doi.org/10.1007/978-3-8348-2269-7_7.

[22] B. Lenze, „Einwegfunktionen," in *Basiswissen Angewandte Mathematik – Numerik, Grafik, Kryptik: Eine Einführung mit Aufgaben, Lösungen, Selbsttests und interaktivem Online-Tool*. Wiesbaden: Springer Fachmedien Wiesbaden, 2020, S. 253–261, ISBN: 978-3-658-30028-9. DOI: 10.1007/978-3-658-30028-9_14. Adresse: https://doi.org/10.1007/978-3-658-30028-9_14.

[23] A. Beutelspacher, H. B. Neumann und T. Schwarzpaul, „Der diskrete Logarithmus, Diffie-Hellman-Schlüsselvereinbarung, ElGamal-Systeme," in *Kryptografie in Theorie und Praxis: Mathematische Grundlagen für Internetsicherheit, Mobilfunk und elektronisches Geld*. Wiesbaden: Vieweg+Teubner, 2010, S. 132–144, ISBN: 978-3-8348-9631-5. DOI: 10.1007/978-3-8348-9631-5_11. Adresse: https://doi.org/10.1007/978-3-8348-9631-5_11.

[24] C. Karpfinger und H. Kiechle, „Diskrete Logarithmen," in *Kryptologie: Algebraische Methoden und Algorithmen*. Wiesbaden: Vieweg+Teubner, 2010, S. 179–190, ISBN: 978-3-8348-9356-7. DOI: 10.1007/978-3-8348-9356-7_10. Adresse: https://doi.org/10.1007/978-3-8348-9356-7_10.

[25] C. Karpfinger und H. Kiechle, „Die Verfahren von Diffie und Hellman, ElGamal und Rabin," in *Kryptologie: Algebraische Methoden und Algorithmen*. Wiesbaden: Vieweg+Teubner, 2010, S. 167–178, ISBN: 978-3-8348-9356-7. DOI: 10.1007/978-3-8348-9356-7_9. Adresse: https://doi.org/10.1007/978-3-8348-9356-7_9.

[26] D. A. Carts, „A review of the Diffie-Hellman algorithm and its use in secure internet protocols," *SANS institute*, S. 1–7, 2001.

[27] J.-F. Raymond und A. Stiglic, „Security issues in the Diffie-Hellman key agreement protocol," *IEEE Transactions on Information Theory*, Jg. 22, S. 1–17, 2000.

[28] N. Mäurer, T. Gräupl, C. Gentsch und C. Schmitt, „Comparing different Diffie-Hellman key exchange flavors for LDACS," in *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*, IEEE, 2020, S. 1–10.