Kryptologie – Methoden, Anwendungen und Herausforderungen

Claus Diem

Zusammenfassung. Die Verarbeitung von Informationen durch elektronischer Geräte führt zu einer Vielzahl sicherheitsrelevanter Herausforderungen. Mittels Kryptographie kann vielen dieser Herausforderungen begegnet werden und neue Anwendungen werden möglich gemacht.

Welche Methoden werden hierbei eingesetzt? Auf welchen mathematischen Grundlagen beruhen diese? Wie sind die heute vorherrschenden Ideen und Methoden entstanden? Welche aktuelle Entwicklungen, welche Herausforderungen gibt es oder deuten sich an und wie könnte auf diese reagiert werden?

2010 Mathematics Subject Classification. Primary 01-A67; Secundary 01-A65, 94-03, 11-03

1. Verschlüsselte Nachrichten und mehr

Geheimagenten und online Kaufhäuser benutzen sie ebenso wie Schüler, die „geheime Nachrichten“ mit merkwürdigen Symbolen austauschen: die Kryptographie. Das Wort „Kryptographie“ ist abgeleitet aus den beiden griechischen Wörtern „κρυπτος“, ‚verborgen‘, und „γραφειν“, ‚schreiben‘, es geht also um Geheimschrift.

Der Sender *verschlüsselt* die Nachricht der Empfänger *entschlüsselt* sie mittels eines zuvor vereinbarten Geheimnisses. Das Bemühen, verschlüsselte Botschaften zu lesen, ohne das vereinbarte Geheimnis zu kennen, wird *Kryptoanalyse* genannt. Es ist üblich, beide Gesichtspunkte unter *Kryptologie* zusammen zu fassen.

Aufgrund der technischen Entwicklung im Bereich der Elektronik werden die Begriffe Kryptographie und Kryptologie heutzutage weiter gefasst; die Ziele der Kryptographie nach heutigem Sprachgebrauch umfassen alle sicherheitsrelevanten Aspekte des Verarbeitens, Übertragens und Benutzens von Information in Anwesenheit eines Gegners.

Kryptographische Verfahren sind hierbei auf viele Gebiete vorgedrungen. Man kann sie benutzen, um *Vertraulichkeit* bei jeder Art elektronischer Kommunikation herzustellen. Sie werden zur *Authentisierung* benutzt, beispielsweise beim Aufschließen eines Autos und Lösen einer Wegfahrsperre, beim Abheben von Geld an einem Bankterminal mittels einer Chipkarte oder beim sich Ausweisen an einer Grenze mittels eines Reisepasses. Dokumente werden heutzutage oftmals mittels kryptographischer Methoden digital signiert, zum Beispiel durch einen Notar; hiermit wird die *Verbindlichkeit* von Vereinbarungen sichergestellt. Mittels elektronischer Signaturen kann auch die *Integrität* von Dokumenten sichergestellt werden;

dies wird zum Beispiel in Passen angewandt.

Wie stellt sich die Kryptologie in einer Welt mit elektronischen Geräten, in der Datenerfassung und -verarbeitung stetig zunehmen, dar? Welche Grundlagen, welche Anwendungen hat sie? Wie entwickelten sich die vorherrschenden Ideen und Methoden historisch und welche Herausforderungen gibt es?

2. Klassische Ideen

Bis zum Ende des ersten Weltkriegs entwickelte sich die Kryptologie langsam, die verwendeten Verfahren waren vom heutigen Gesichtspunkt aus elementar. Die verwendeten Begriffe und viele Ideen sind auch heute noch grundlegend.

2.1. Vorgehensweisen. Zunächst zwei Begriffe: Bei Benutzung eines kryptographischen Verfahrens werden Texte verschlüsselt, versendet, empfangen und dann wieder entschlüsselt. Der ursprüngliche Test heißt *Klartext* und der verschlüsselte Text *Chiffriertext*.

Klassischerweise gibt es zwei grundsätzlich verschiedene Vorgehensweisen, um Texte zu verschlüsseln: Erstens kann man quasi auf atomarer Ebene ansetzen, nämlich auf der Ebene der Symbole (Buchstaben, Zahlzeichen, Satzzeichen). Zweitens kann man auf der Ebene der Wörter ansetzen.

Der zweite Ansatz ist leichter zu fassen, weil er von der Methode her weniger Variationsmöglichkeiten bietet: Mittels eines speziellen *Codebuches* werden hier Wörter der Standardsprache durch Code-Wörter ersetzt. Diese Code-Wörter können dabei andere Wörter der Standardsprache oder auch beliebige Kombinationen von Buchstaben und Zahlzeichen sein.

Für den ersten Ansatz wurde über die Jahrhunderte eine enorme Anzahl von Verfahren entwickelt. Zwei naheliegende Ideen kommen dabei aber immer wieder vor: die *Substitution* und die *Transposition*.

Bei der Substitution werden die einzelnen Symbole (Buchstaben, Zahlzeichen, das Leerzeichen, Satzzeichen) durch andere Symbole (oder Kombinationen von Symbolen) ersetzt. Diese anderen Symbole können ausgedacht oder auch normale Buchstaben oder Zahlzeichen sein. Wenn es auch naheliegend ist, „mysteriöse“ Symbole zu verwenden, macht dies für die Sicherheit keinen Unterschied.

Bei der Transposition wird die Reihenfolge der Zeichen in Text nach einer bestimmten Regel vertauscht. Ein gutes Beispiel hierfür ist: Man schreibe den Text zeilenweise in eine Tabelle von links oben nach rechts unten. Dann lese man die Spalten der Tabelle in einer vorher festgelegten Reihenfolge von oben nach unten ab.

Selbstverständlich kann man diese Verfahren nun miteinander kombinieren. Man könnte zum Beispiel zunächst ein Codebuch verwenden und dann eine Transposition und auch noch eine Substitution.

Klassische kryptographische Verfahren beruhen auf gemeinsamen Geheimnissen zwischen Sender und Empfänger. In aller Regel bieten die Verfahren die Möglichkeit, Texte mit variablen zu vereinbarenden Geheimnissen zu verschlüsseln. Beim

Codebuchverfahren ist das variable Geheimnis das Codebuch selbst, beim Substitutionsverfahren die Substitutionstabelle und beim beschriebenen Transpositionsverfahren die Anzahl und die Reihenfolge der Spalten.

Dieses Geheimnis nennt man den *Schlüssel*. Man kann also – zumindest in den genannten Beispielen – zwischen dem Verfahren selbst und dem Schlüssel unterscheiden. Wir werden sehen, dass diese Unterscheidung von besonderer Wichtigkeit ist.

2.2. Der Wettlauf der Kryptologie. Da niemand ein unsicheres kryptographisches Verfahren verwenden möchte, werden potentielle Verfahren schon im Vorfeld auf mögliche Angriffe getestet und Verfahren, die benutzt werden, ständig überprüft.

Es entsteht so ein Wettlauf zwischen Entwerfen neuer Verfahren und Angriffen auf dieselben. Diesen Wettlauf vollziehen wir hier exemplarisch am klassischen Substitutionsverfahren nach. Die folgende Beschreibung ist eher idealtypisch als historisch, wenn es auch über den Lauf der Jahrhunderte entsprechende Überlegungen und Entwicklungen gegeben hat.

Beim Substitutionsverfahren, so wie es beschrieben wurde, fällt sofort auf: Manche Symbole, wie beispielsweise das `E` oder das Leerzeichen, kommen in normalen Sprachen deutlich häufiger als andere vor. In der Regel gilt dies dann auch für einen bestimmten Klartext. Man kann dann davon ausgehen, dass das in einem Chiffriertext am häufigsten vorkommende Symbol wohl dem `E` oder dem Leerzeichen entspricht. Einen Text in einer bekannten Sprache, der ein paar Zeilen lang ist, kann man in der Regel mittels Buchstabenhäufigkeiten rekonstruieren.

Das Verfahren kann als gebrochen gelten. An dieser Stelle ist es naheliegend zu fragen: Gibt es eine Variante, auf die zumindest der angegebene Angriff nicht anwendbar ist?

Ja, die gibt es: Man muss dafür sorgen, dass im Chiffriertext alle Symbole ungefähr gleich häufig vorkommen. Diese Methode beruht auf der Idee, dass ein Symbol zu mehreren verschieden Symbolen verschlüsselt werden kann. Dies wird so angewandt, dass aller Wahrscheinlichkeit nach in einem Chiffriertext die Symbole alle etwa gleichhäufig vorkommen. Nehmen wir zum Beispiel an, dass 1000 verschiedene Symbole für das Schreiben der Chiffriertexte verwendet werden. Wenn nun der Buchstabe `E` mit einer Wahrscheinlichkeit von 12,8 % in einer Sprache vorkommt, werden ihm 128 verschiedene Symbole zugeordnet. Beim Verschlüsseln wird für jedes Vorkommen von `E` eines der 128 Symbole ausgewählt und geschrieben. Hiermit läuft die oben beschriebene Häufigkeitsanalyse ins Leere. Aber wo sollen 1000 Symbole herkommen? Nun, dies ist einfacher als man zunächst denken könnte: Man startet mit den 10 Zahlzeichen von `0` bis `9` und betrachtet jede Kette von drei Zahlzeichen als eigenständiges Symbol. Im Konkreten werden dann dem Buchstaben `E` 128 Ketten von je drei Zahlzeichen zugeordnet.

Nachdem das Verschlüsselungsverfahren modifiziert worden ist, ist es naheliegend zu fragen, ob die Angriffsmethode auch verändert werden kann. Ja, dies ist möglich: Man betrachtet nun nicht mehr die Häufigkeiten von Symbolen in der Sprache und im Chiffriertext (im konkreten Beispiel im Chiffriertext gilt hier eine

Kette von drei Ziffern als ein Symbol), sondern von sogenannten *Bigrammen*, das sind Kombinationen von zwei Symbolen, oder von *Trigrammen*, also Kombinationen von drei Symbolen. Nun kommen einige Bigramme und Trigramme deutlich häufiger vor als andere.

Erneut ist das Verfahren gebrochen und es stellt sich die Frage, ob dieses Verschlüsselungsverfahren nochmals verbessert werden kann oder ob man vielleicht eine vollkommen andere Methode verwenden sollte.

Eine naheliegende Antwort auf diese Frage ist, dass das Konzept der Substitution so grundlegend ist, dass es in jedem Fall ein Teil der Methode sein sollte. Man könnte beispielsweise eine Substitution mit einer Transposition verbinden.

Bis jetzt haben die betrachteten Angriffe keine Informationen über die gesendeten Nachrichten selbst verwendet, solche Informationen können aber auch benutzt werden. Beispielsweise beginnen und enden Briefe oftmals mit standardisierten Grußformeln und militärische Nachrichten sind oftmals stark strukturiert. Wenn dies nicht weiterführt, könnte ein Angreifer auch versuchen, den Gegner eine ihm untergejubelte Nachricht verschlüsseln zu lassen, so dass er dann zu einem vorgegebenen Klartext den entsprechenden Chiffriertext hat. Es könnte auch möglich sein, dass ein Angreifer mittels eines oder mehrerer solcher Paare von Klar- und Chiffriertext gesuchte Informationen aus einer anderen geheimen Nachricht extrahieren kann, ohne überhaupt den Schlüssel zu finden.

3. Neue Ideen

Nach der technischen Entwicklung lässt sich die Geschichte der Kryptologie in drei Perioden einteilen:

1. Das *Papier-und-Bleistift Zeitalter* bis etwa zum Ende des I. Weltkriegs.
2. Das *Zeitalter der elektrisch-mechanischen Chiffriermaschinen* von etwa dem Ende des I. Weltkriegs bis etwa 1970.
3. Das *elektronische Zeitalter* ab etwa 1970.

Die erste Periode der Kryptologie war dadurch gekennzeichnet, dass für die Kryptographie höchstens einfache mechanische Geräte zum Einsatz kamen. Diese Geräte konnten des weiteren meist durch Tabellen und einfache Rechnungen ersetzt werden. Für die Kryptoanalyse wurden neben ad hoc Ansätzen hauptsächlich statistische Methoden eingesetzt, etwa so wie in Abschnitt 2.2 beschrieben.

In der zweiten Periode wurden für die Kryptographie neben handschriftlichen Methoden elektrisch-mechanische Maschinen wie die deutsche Enigma eingesetzt. In der Kryptoanalyse wurden mit der Zeit Methoden entwickelt, die über rein statistische Methoden hinausgingen, und hierfür wurden dann auch elektrische Rechenmaschinen / Computer eingesetzt.

Das elektronische Zeitalter begann mit dem Aufkommen der Datenverarbeitung. Da die auch zu Beginn entwickelten Methoden und Verfahren immer noch

verwendet werden oder die jetzigen Verfahren eine direkte Weiterentwicklung darstellen, kann es als die Gegenwart der Kryptologie gelten. Es zeichnet sich nicht nur durch die verwendete Technik sondern auch durch das Streben nach wissenschaftlichen Methoden in der Kryptographie, einer starken Verbindung zur Mathematik und einer hohen Innovationsgeschwindigkeit aus. In diesem Abschnitt wollen wir hauptsächlich die Entwicklung in dieser nahen Vergangenheit nachvollziehen, wobei die konzeptionellen Ideen im Vordergrund stehen.

3.1. Vier Texte. Auf vier wegweisende Texte reduzieren wir die ersten beiden Perioden. Diese reduzierte Darstellung wird der Geschichte der Kryptologie vor dem elektronischen Zeitalter selbstredend nicht gerecht und ein Leser, der sich mehr für die ersten beiden Perioden interessiert, sei aufgerufen, das Standardwerk der Geschichte der Kryptologie zu lesen, das erstmals 1967 erschienene Buch *The codebreakers* von David Kahn mit Neuauflage im Jahr 1996 ([18]). Für neuere Ergebnisse der Forschung zur Geschichte der Kryptologie sei das Buch *Codeknacker und Codemacher* von Klaus Schmeh ([30]) empfohlen.

3.1.1. رسالة أبي يوسف يعقوب بن إسحاق الكنديّ في استخراج المعمّى إلى أبي العبّاس (Abū Yūsuf Ya'qūb ibn Isḥāq al-Kindī: Das Sendschreiben über Kryptoanalyse an Abū l-'Abbās). Die erste *systematische Darstellung* der Kryptologie entstammt nach heutigem Wissen aus dem islamischen Mittelalter. Der Autor ist der aristotelische Philosoph al-Kindī, der im 9. Jahrhundert u.Z. in Bagdad gelebt hat.

Der Text galt wie auch zwei Texte über Kryptologie aus dem 13. und 14. Jahrhundert lange Zeit als verloren. In den 1980er Jahren wurden für jeden der Texte jeweils ein Manuskript gefunden, welche daraufhin editiert und veröffentlicht wurden.

In dem Manuskript, das wohl al-Kindīs Text enthält, werden verschiedene Verfahren basierend auf Substitution und Transposition von Buchstaben erörtert; die Verfahren werden sogar in einem Baum-Diagramm klassifiziert. Die Kryptoanalyse wird auf der Basis der Häufigkeitsanalyse entwickelt, wobei auch Bi- und Trigramme betrachtet werden.

Übrigens: Das französische Wort „Chiffre“ bedeutet sowohl Verschlüsselungsverfahren als auch Ziffer, und sowohl „Chiffre“ als auch „Ziffer“ stammen vom arabischen Wort für Null „صفر“ („ṣifr“) ab. Diese und ähnliche Wörter europäischer Sprachen haben im Lauf der Jahrhunderte Null, Ziffer, Chiffrierverfahren und Chiffriertext bedeutet.

3.1.2. Leon Battista Alberti: De Componendis Cifris (Über das Schreiben in Chiffren). Aus dem in Kleinstaaten zerfallenen Italien stammt der älteste überlieferte europäische Text zur Kryptologie, geschrieben ungefähr im Jahr 1466. Sein Autor Leon Battista Alberti gilt als die Verkörperung des „universellen Renaissancemenschen“; er arbeitete als Architekt und schrieb bemerkenswert viele Texte zu den unterschiedlichsten Themen und auch literarische Werke.

Alberti propagiert die Idee, bei der Substitution von Buchstaben die Verschlüsselungstabelle während der Verschlüsselung zu wechseln. Hierzu erfand er, wie es scheint, die *Chiffrierscheiben*, die bis ins 19. Jahrhundert populäre Geräte für Kryptographie waren. Diese Geräte bestehen aus zwei übereinanderliegenden und in der Mitte zusammengenieteten Scheiben, wobei die untere Scheibe größer ist. Auf den Rändern ist jeweils ein Alphabet geschrieben. Für jede Position der beiden Scheiben zueinander erhält man somit eine bestimmte Substitution.

Für das von Alberti ersonnene Verfahren ist die Ordnung der Alphabete permutiert und die Stellung der Scheiben wird nach einigen Wörtern verändert. Zusätzlich schlug Alberti die Benutzung eines Codebuchs für die wichtigsten Wörter vor.

Die wohl älteste erhaltene Chiffierscheibe, eine französische Scheibe von der Zeit Ludwig XIV.
Quelle: Nicholas Gessler Sammlung

Dieses konkrete Verfahren war jedoch wohl nicht viel in Gebrauch. Stattdessen wurde ein viel einfacheres aber wesentlich unsicheres Verfahren unter dem Namen *Vigenère-Verfahren* bei Laien populär. Für dieses Verfahren sind beide Alphabete in identischer (üblicher) Reihenfolge und die Position wird nach jedem Buchstaben geändert. Das gemeinsame Geheimnis ist nun ein Schlüsselwort (Codewort). Wenn dieses beispielsweise `BUCH` ist, so wird zuerst das `A` auf `B` gedreht, dann auf das `U`, dann auf das `C` und schließlich auf das `H`, wonach wieder von vorne begonnen wird. Professionelle Kryptographen wussten jedoch, dass dieses Verfahren ziemlich schwach war und benutzten ausgefeiltere Verfahren, die Codebücher involvierten. (Für mehr Informationen hierzu siehe Kapitel 4 von [18].)

3.1.3. Auguste Kerckhoffs: La cryptographie militaire. „Diese Arbeit ([19]) von 1883 ist wohl diejenige Arbeit der Papier-und-Bleistift-Periode, auf die heutzutage am meisten Bezug genommen wird. Der in Frankreich lebende niederländische Linguist und Kryptograph Kerckhoffs stellt in der aus der Sicht Frankreichs geschriebenen Arbeit aus dem Jahr 1883 sechs Forderungen an die militärische Kryptographie auf, die später als die „Kerckhoffs'schen Prinzipien“ bekannt wurden. Wegweisend sind insbesondere die ersten drei Forderungen. In etwas zugespitzter und überarbeiteter Form lauten diese:

1. Das Verfahren muss de facto, wenn nicht mathematisch, unbrechbar sein.

2. Die Bedienung des Verschlüsselungssystems darf kein Geheimnis erfordern und so ein System muss ohne Nachteil in die Hände des Feindes fallen können.

3. Der Schlüssel muss ohne Zuhilfenahme von schriftlichen Notizen übertragen und im Gedächtnis behalten werden können und er muss nach Belieben der Korrespondierenden ausgetauscht oder geändert werden können.

Zur ersten Forderung führt Kerckhoffs aus: Es würde allgemein angenommen, dass es im Krieg ausreiche, wenn ein Chiffrierverfahren Sicherheit für drei bis vier Stunden biete. Es gebe aber sehr wohl Informationen, die über den Tag hinaus von Wichtigkeit seien. „Ohne alle denkbaren Möglichkeiten aufzuzählen", verweist er auf die Kommunikation aus einer belagerten Stadt heraus. Dass ein gutes kryptographisches Verfahren unabhängig von allen Eventualitäten Sicherheit gewährleisten sollte, ist eine Idee, die die Kryptographie seitdem prägt.

Die Argumentation zur zweiten Forderung ist: Kryptographie erfordere immer Geheimnisse. Armeen seien aber so groß, dass man davon ausgehen müsse, dass der Feind jedes Geheimnis kennt, welches eine große Anzahl der eigenen Soldaten kennt. Deshalb dürfen nur sehr wenige Menschen das Geheimnis kennen. Kerckhoffs argumentiert somit für eine Trennung des Verfahrens, das öffentlich sein kann und sogar sollte, von einem kleinen Geheimnis, dem Schlüssel. Dies bedeutet insbesondere, dass keine umfangreichen, schwer geheimzuhaltenden Codebücher verwendet werden sollten. Diese Forderung wird noch vom dritten Prinzip bestätigt.

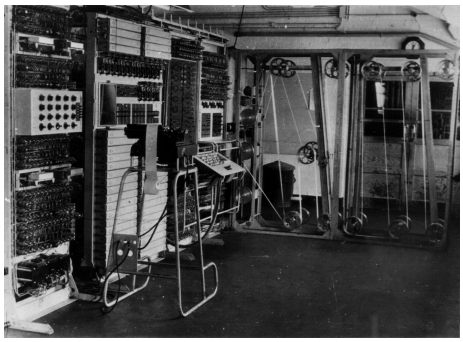
Die aufgestellten Prinzipien und ihre Begründung waren wegweisend für die Entwicklung der Kryptographie, insbesondere für die Etablierung wissenschaftlicher Methoden.

Wie schwierig es allerdings ist, die Prinzipien auch umzusetzen, kann man an Kerckhoffs' Arbeit selbst sehen: Kerckhoffs geht zum Schluss einige ihm bekannte Verschlüsselungsverfahren durch und schlägt eines vor, das zwar praktisch, aber leider auch unsicher ist.

3.1.4. Claude Shannon: Communication Theory of Secrecy Systems. Für die Entwicklung der Kryptologie war diese im Jahr 1948 erschiene Arbeit in der Länge eines kleinen Buches ähnlich bedeutend wie die von Kerckhoffs. Der US-Amerikaner Claude Shannon entwickelt in ihr eine abstrakte mathematische Theorie von Ver- und Entschlüsselung sowie von Kryptoanalyse. Wie kein anderer zuvor bettet er hierbei die Kryptologie in die Mathematik ein. So beschreibt er beispielsweise die Verschlüsselung eines Systems als eine Funktion, die von zwei Variablen abhängt, den Schlüssel und der Nachricht. Die Resultate werden dann als mathematische Sätze formuliert und bewiesen.

Shannon fragt, wie viel Information ein Chiffriertext über den Klartext liefert, gegeben, dass ein Angreifer schon eine gewisse Information hat. Er verwendet hierbei stochastische und statistische Methoden sowie die von ihm entwickelte Informationstheorie. Er führt den Begriff der *perfekten Sicherheit* ein, der beschreibt, dass ein Angreifer durch das Abfangen einer Nachricht keine neuen Informationen enthält. Sodann beweist er, dass perfekte Sicherheit erfordert, dass der Schlüssel als Wort geschrieben mindestens so lang ist wie die zu übermittelnde Nachricht. Dies zeigt auf, dass perfekte Sicherheit so wie von Shannon definiert inkompatibel mit den Kerckhoffsschen Prinzipien ist.

Shannon führt auch zwei Ideen ein, die als Konstruktionsprinzipien von Chiffriersystemen relevant wurden: *Diffusion* und *Konfusion*. In beiden Fällen ist es das Ziel, die Verwendung statistischer Methoden in der Kryptoanalyse zu erschweren. Diffusion bedeutet, dass sich kleine Änderungen des Klartextes auf einen großen Bereich des Chiffriertextes auswirken. Konfusion, so wie von Shannon definiert, bedeutet, dass die Beziehung zwischen dem Chiffriertext und dem Schlüssel eine komplexe ist.Letztere Anforderung wurde später so erweitert, dass auch die Beziehung zwischen Chiffriertext und Klartext komplex ist.



Der Colossus Computer
Quelle: British National Archive

3.2. Das elektronische Zeitalter. Schon während des II. Weltkriegs bauten die Briten erste programmierbare Rechenmaschinen, um die deutschen Verschlüsselungsmaschinen zu knacken, was auch gelang. Für die Kryptographie (im Gegensatz zur Kryptoanalyse) wurden Computer allerdings erst mit dem Aufkommen der elektronischen Datenverarbeitung in den 1960er Jahren relevant.

Die neue Technik ging schnell auch mit weiteren Veränderungen einher: Der Umgang und auch die Forschung mit Kryptographie wurde offener, der Begriff der Kryptographie wurde weiter gefasst und es wurden in ganz neuer Weise mathematische Konzepte eingesetzt. Diese Entwicklung hält bis heute an.

Ein Charakteristikum von Computern ist das Speichern aller Daten als Ketten von bits; die bits spielen nun die Rolle von Atomen, genau wie dies die Buchstaben im Papier-und-Bleistift Zeitalter taten. Es ist nun naheliegend, eine Verschlüsselung auf dieser Ebene anzusetzen, was auch alle im Folgenden diskutierten Verfahren tun. Wir erinnern auch daran, dass man natürliche Zahlen (welche nach Definition auch die 0 einschließen sollen) im Zweiersystem, das heißt mit Ketten von bits ausdrücken kann. Beispielsweise wird die Zahl 10 durch die bit-Kette `IOIO` dargestellt. Die Anzahl der notwendigen bits, um eine Zahl auf diese Weise darzustellen, wird die *bit-Länge* der Zahl genannt; für eine Zahl $a \neq 0$ ist diese ungefähr $\log_2(a)$. Darüberhinaus kann man, indem man vor solchen Ketten Nullen anfügt, die bit-Ketten der Länge ℓ mit den natürlichen Zahlen kleiner als 2^ℓ identifizieren.

3.2.1. Vom Data Encryption Standard zum Advanced Encryption Standard. Im Jahr 1973 machte die Vorläuferinstitution des US-amerikanischen National Institute of Standards and Technology (NIST) eine Ausschreibung für einen zu standardisierenden elektronischen Chiffrieralgorithmus, wobei ein *Algorithmus* ein Rechenverfahren ist. Nachdem zuerst kein eingereichtes Verfahren als adäquat betrachtet wurde, wurde 1977 eine Variante eines von IBM entwickeltes Verfahrens zum *Data Encryption Standard (DES)* erhoben. Wie alle Standards von NIST

galt auch dieser Standard offiziell nur für die amerikanische Regierung und ihre Vertragspartner. Er wurde aber schnell zum de facto Industriestandard – was zu erwarten war.

Mit dem Verfahren werden Blöcke bestehend aus 64 bit verschlüsselt. Das Verfahren besteht aus 16 Runden, in welchen immer wieder nach demselben Schema vorgegangen wird. In jeder Runde wird zunächst ein Teil des Schlüssels mittels exklusiv-oder (XOR) bitweise mit einem Teil des aktuellen Zwischenergebnisses verbunden. Danach wird das Zwischenergebnis in kleine Blöcke von 6 bit aufgeteilt. Mittels festen aber „zufällig aussehenden“ Tabellen wird jeder der 6 bit Langen bit-Ketten durch eine andere bit-Kette ersetzt. Schließlich werden die Blöcke untereinander transponiert.

Man kann also sagen, dass das Verfahren auf einer wiederholten Anwendung der Ideen Shannons beruht: Mittels der Tabellen erhält man Konfusion und mittels der Transpositionen erhält man Diffusion.

Es wurde bis zum heutigen Tag kein praxisrelevanter Angriff auf das Verfahren gefunden, der schneller als reines Ausprobieren ist. Ein Problem ist jedoch die Schlüssellänge von nur 56 bit, die schon von Beginn an von Kryptographen wie Whitfield Diffie und Martin Hellman (die in Abschnitt 3.2.4 eine entscheidende Rolle spielen werden) als zu kurz kritisiert wurde. Durch die Entwicklung der Computer wurde die Schlüssellänge dann wirklich untragbar kurz. NIST führte deshalb im Jahr 1997 einen offenen Wettbewerb für ein neues Verfahren durch, das unter *Advanced Encryption Standard (AES)* bekannt werden sollte. Die Beteiligung war nun lebhaft und zum Advanced Encryption Standard erhoben wurde ein Verfahren von zwei belgischen Kryptographen.

Das Prinzip dieses Verfahrens ist ähnlich wie das von DES: Wiederum werden Blöcke abgearbeitet und das Verfahren arbeitet in Runden, in denen jeweils der Schlüssel eingespeist und für Konfusion und Diffusion gesorgt wird. Im Vergleich zu DES beruht das Verfahren auf mathematisch klareren und eleganteren Konstruktionen. Durch die klare Konstruktion kann bewiesen werden, dass bestimmte potentielle Angriffe ausgeschlossen sind.

Erwartungsgemäß ist gegen AES als mathematisches Verfahren bis jetzt kein relevanter Angriff gefunden worden. Mit einer vorgeschriebenen Schlüssellänge von mindestens 128 bit scheint das Verfahren optimalen Schutz für viele Jahrzehnte zu bieten. Wie so oft in der Kryptographie muss man hier aber aufpassen, keine voreiligen Schlüsse zu ziehen. Es ist nämlich möglich, dass ein Angreifer auf überraschende Art interne Informationen über die ausgeführten Rechnungen und somit vielleicht auch über den Schlüssel erhält. So gibt es gegen „straight-forward“-Implementierungen von AES mehrere praxisrelevante Angriffe, die auf Analyse der Rechenzeit beruhen (siehe [5]).

3.2.2. Passwort-Verschlüsselung. Die Entwicklung von Multinutzer-Computern führte zu einem Problem: Wenn jeder Benutzer ein Passwort hat, wie können dann alle diese Passwörter gegen Spionage geschützt werden? Wie kann insbesondere sichergestellt werden, dass der Systemadministrator, der Zugriff auf das gesamte System hat, die Passwörter nicht einsehen kann?

Nehmen wir an, wir haben eine Funktion f, die jedem Passwort ein „verschlüsseltes Passwort“ zuordnet und die folgenden Eigenschaften hat: Erstens ist f schnell zu berechnen und zweitens ist es praktisch unmöglich, zu einem verschlüsselten Passwort C ein echtes Passwort P zu finden, das mittels f auf C abgebildet wird, d.h. mit $f(P) = C$. Dann erhält man das folgende Authentisierungsverfahren:

Anstatt des Passworts P eines bestimmten Benutzers, sagen wir Ursa, zu speichern, speichert man $C = f(P)$. Wenn nun ein Benutzer ein Passwort P' eingibt, überprüft man, ob $f(P) = f(P')$ ist. Wenn dies der Fall ist, ist der Benutzer als Ursa authentisiert.

Die Idee wurde 1967 von Roger Needham an der Universität Cambridge entwickelt und verwirklicht und von seinem Kollegen Maurice Wilkies *one-way cipher*, d.h. *Einweg-Verschlüsselungsverfahren*, genannt ([34]).

Auf Grundlage eines Verschlüsselungsverfahrens wie DES oder AES kann man die Idee wie folgt realisieren: Man wendet ein Verschlüsselungsverfahren auf P als Schlüssel und einen konstanten Klartext wie beispielsweise `0...0` an; das Ergebnis ist dann $f(P)$. Wenn das Verschlüsselungsverfahren sicher ist, erhält man eine Funktion mit den gewünschten Eigenschaften.

Neben der Anwendung selbst ist die Idee aus zwei Gründen interessant: Erstens wird hier verdeutlicht, dass es eine tiefe Beziehung zwischen dem klassischen Ziel der Kryptographie, der Vertraulichkeit, mit anderen Zielen wie dem der Authentisierung gibt. Zweitens liefert die Idee einer Funktion f wie soeben beschrieben eine Verbindung zur *Komplexitätstheorie*.

3.2.3. Komplexitätstheorie. Durch die Entwicklung der Computer bildete sich Ende der 1960er Jahre eine neue Disziplin heraus, die *Informatik*. Innerhalb der Informatik wiederum entwickelte sich die *Theoretische Informatik*. Dieses Gebiet umfasst die theoretische Beschäftigung mit formalen Rechenmethoden oder *Algorithmen*. Von wissenschaftlichen Standpunkt aus fällt das Gebiet der Theoretischen Informatik vollständig in die Mathematik.

Ein wichtiger Teilbereich der Theoretischen Informatik ist die *Komplexitätstheorie*. Hier werden Fragestellungen der folgenden Form untersucht: Gegeben sei ein rechnerisches Problem, beispielsweise die Addition oder die Multiplikation natürlicher Zahlen oder das Problem, natürliche Zahlen zu faktorisieren. Wie schnell können die Rechnungen für immer größer werdende Zahlen mittels eines Algorithmus für eine idealisierte elementare Rechenmaschine durchgeführt werden? Hier werden also nicht Rechnungen für konkrete Eingaben (man spricht hier von *Instanzen*) betrachtet und auch nicht Rechnungen für Eingaben einer bestimmten Größenordnung, sondern vielmehr alle denkbaren Rechnungen für alle (unendlich vielen) Eingaben. Man stellt sich hierbei vor, dass die idealisierte Rechenmaschine bitweise operiert und man misst ihre Laufzeit entsprechen. Wir führen dies anhand der genannten Beispiele aus, beginnend mit der Addition von zwei natürlichen Zahlen.

Die beiden Zahlen seien wie zu Beginn von Abschnitt 3.2 ausgeführt in binärer Darstellung gegeben. Die Laufzeit soll in der Eingabelänge ausgedrückt werden, die wir mit ℓ bezeichnen. (ℓ ist in etwa $\log_2(a) + \log_2(b)$ für $a, b \neq 0$.) Mit der

schriftlichen Addition erhält man nun eine Laufzeit von höchstens $C \cdot \ell$ für eine Konstante $C > 0$.

Analog betrachten wir das Problem, zwei Zahlen a und b zu multiplizieren. Mit der schriftlichen Multiplikation benötigt man nun nicht mehr als ℓ Additionen von Zahlen einer Eingabelänge von höchstens 2ℓ; man erhält eine Laufzeit von höchstens $C' \cdot \ell^2$ für eine Konstante $C' > 0$.

Man drückt dies so aus: Die obere Schranke für die Laufzeit der Addition ist *linear* in ℓ und die obere Schranke für die Laufzeit der Multiplikation ist *quadratisch* in ℓ. Nun gibt es auch noch andere Methoden für die Multiplikation als die übliche schriftliche Multiplikation. So gibt es eine Methode, mit denen man eine Laufzeit von $C'' \cdot \ell^{1,5}$ mit einer Konstante $C'' > 0$ erzielen kann. Diese obere Schranke ist von einer bestimmten Größe ℓ an besser (d.h. kleiner) als die klassische quadratische. Mit anderen Worten: Die Schranke, die man mit der alternativen Methode erhält, ist für alle außer endlich viele Eingaben besser als die klassische.

Man sagt, dass die Schranke, die man mit der alternativen Methode erhält, *asymptotisch* besser ist. In der Komplexitätstheorie stehen nun solche asymptotischen Bewertungen im Vordergrund und im Sinne der Komplexitätstheorie gilt das alternative Verfahren dann auch als schneller. Dies sagt allerdings nichts darüber aus, welche Methode für KONKRETE Zahlen schneller ist. Solche Aussagen werden in der Komplexitätstheorie in der Regel nicht untersucht.

Eine Laufzeit, die sich durch $C \cdot \ell^k$ für gewissen $C, k > 0$ nach oben abschätzen lässt, heißt *polynomiell*. In der Komplexitätstheorie gelten nun Algorithmen mit polynomieller Laufzeit als „schnell“ in einem qualitativen Sinn und werden einfach als „schnell“ oder als „effizient“ bezeichnet. Um den komplexitätstheoretischen Ansatz hervorzuheben, benutzen wir den Ausdruck „qualitativ schnell“ und modifizieren die anderen Begriffe der Komplexitätstheorie entsprechend.

Wir sehen, dass die Probleme, zwei natürliche Zahlen zu addieren oder zu multiplizieren, polynomielle Laufzeit hat und somit qualitativ schnell im soeben definierten Sinn ist.

Betrachten wir nun hingegen das Problem, eine (natürliche) Zahl zu faktorisieren, wobei die Zahl wieder variabel ist. Es ist keine Methode bekannt, mit der man dieses Problem qualitativ schnell, d.h. in polynomieller Laufzeit lösen kann. Dies gilt auch, wenn man zulässt, dass die Algorithmen während der Rechnung „würfeln“ dürfen, eine Operation, die wir von nun an erlauben wollen, wenn wir von Algorithmen reden. Es ist sogar kein randomisierter Algorithmus bekannt, mit dem man für einen „nicht-vernachlässigbaren“ Anteil von Produkten pq von Primzahlen p, q gleicher bit-Länge die Faktorisierung berechnen kann, wobei der Begriff „nicht-vernachlässigbar“ auch präzise definiert werden kann. Mehr noch, es ist auch kein qualitativ schneller Algorithmus bekannt, der für einen nicht-vernachlässigbaren Anteil von Produkten zweier natürlicher Zahlen derselben bit-Länge eine Faktorisierung in zwei natürliche Zahlen derselben bit-Länge berechnet.

Den beschriebenen Sachverhalt kann man mittels den Begriffen der Komplexitätstheorie so ausdrücken: Betrachten wir die Funktion f, die zwei natürlichen Zahlen m und n ihr Produkt mn zuordnet, d.h. $f(m, n) = mn$. Diese Funktion kann in qualitativ schnell berechnet werden. Hingegen ist kein randomisierter

Algorithmus bekannt, mit dem man für einen nicht-verschwindenden Anteil an Funktionswerten y (dies sind hier Produkte mn) sogenannte *Urbilder* (dies sind hier die Tupel (m, n) mit $mn = y$) qualitativ schnell berechnen kann.

Wenn es wirklich keinen solchen randomisierten Algorithmus gibt, handelt es sich bei der betrachteten Funktion um eine sogenannte *Einweg-Funktion*. Mittels des Konzepts der Einweg-Funktion wird nun das praktische Problem der „Einweg-Verschlüsselung“ wie im vorherigen Abschnitt beschrieben mit der Komplexitätstheorie verbunden.

Der nun vorherrschende komplexitätstheoretische Blick auf die Kryptologie ist ein ganz anderer als derjenige Shannons. Während Shannon der Frage nachging, INWIEWEIT ein Chiffriertext einen Klartext bestimmt, oder inwieweit man aus einem Chiffriertext Informationen über den Klartext berechnen kann, wenn man beliebig viel Rechenleistung zur Verfügung hat, ist der komplexitätstheoretische Gesichtspunkt, ob man in gewisser Weise Klartexte aus Chiffriertexten SCHNELL berechnen kann oder ob dieses Unterfangen als rechnerisch unmöglich zu gelten hat.

Beim Begriff „schnell“ muss man allerdings Vorsicht walten lassen: Wie bereits erwähnt, werden in der Komplexitätstheorie nicht KONKRETE Berechnungen (z.B. eine konkrete Multiplikation oder Faktorisierung) betrachtet. Vielmehr werden qualitative Aussagen über die Geschwindigkeit von Lösungsmethoden für rechnerische Probleme für beliebige (und somit beliebig große) Instanzen, d.h. Eingaben gemacht. Die Aussage, dass eine bestimmte Funktion nun eine Einweg-Funktion ist, ist keine Aussage über rechnerische Schwierigkeit für konkrete Instanzen. Es gibt somit eine LÜCKE zwischen der komplexitätstheoretischen Betrachtung und einer möglichen praktischen Anwendung in der Kryptographie, wie beispielsweise in der Passwortverschlüsselung. Diese Lücke zu schließen, ist keine leichte Aufgabe.

Auch vom theoretischen Gesichtspunkt aus gibt es ein Problem beim grundlegenden Begriff der Einweg-Funktion: Es ist keine einzige Funktion bekannt, von der bewiesen worden ist, dass es sich um eine solche handelt – wenn es auch einige gute Kandidaten für Einweg-Funktionen gibt, wie beispielsweise die soeben beschriebene.

Die Situation ist sogar noch verzwickter: Wenn man von IRGENDEINER Funktion zeigen kann, dass sie eine Einweg-Funktion ist, dann ist das berühmteste offene Problem der Theoretischen Informatik und eines der entscheidendsten Probleme der Mathematik gelöst, das P versus NP Problem. Dieses Problem gilt als eines der schwierigsten offenen Fragen der Mathematik überhaupt. Es ist eines der sogenannten *millenium prize problems* des *Clay Mathematics Instituts*; eine Lösung wird mit einer Million US-Dollar belohnt. Eine einfache – allerdings nicht klassische – Formulierung des Problems ist wie folgt: Gibt es einen Algorithmus, der die Antwort auf die folgende Fragestellung, genannt das *Teilsummenproblem*, in polynomieller Laufzeit berechnet: Gegeben beliebig viele natürliche Zahlen $a_1, \ldots, a_k$ und eine natürliche Zahl S, ist die Summe über EINIGE der Zahlen $a_1, \ldots, a_k$ gleich S? Beispielsweise ist für die vier Zahlen $5; 9; 11; 23$ und $S = 39$ die Antwort „ja“, weil $5 + 11 + 23 = 39$ ist, während sie „nein“ ist, wenn man n zu 38 oder 40 abändert. Nun ist bekannt: Wenn es irgendeine Einweg-Funktion gibt, dann ist

das P versus NP Problem negativ beantwortet, das heißt, es gibt keinen solchen Algorithmus.

3.2.4. Schlüsselaustausch und Kryptographie mit öffentlichen Schlüsseln. Nach Kerckhoffs' Prinzipien sollte ein kryptographisches Verfahren vom Schlüssel unterschieden werden. Das Verfahren sollte allgemein bekannt sein, während selbstredend die einzelnen Schlüssel geheim gehalten werden müssen. Zwei Parteien, die miteinander kommunizieren wollen, können sich also in der Öffentlichkeit auf ein Verfahren einigen. Es erscheint aber einleuchtend, ja sogar selbstevident, dass die beiden Parteien sich nicht auch in der Öffentlichkeit – unter Beobachtung aller möglichen Gegner – auf einen geheimen Schlüssel einigen können.

Dass dies doch geht, wurde 1976 von Whitfield Diffie und Martin Hellman in der Arbeit mit dem wegweisenden Titel „New directions in cryptography" ([7]) gezeigt. Sie stellten ein Verfahren vor, das nun das *Diffie-Hellman Verfahren* heißt.

Mit dem genannten Verfahren können zwei Personen, in der Kryptographie stets *Alice* und *Bob* genannt, in der Öffentlichkeit ein gemeinsames Geheimnis vereinbaren.

Wir stellen das Verfahren kurz vor. Hierzu erläutern wir zunächst das sogenannte *Modulorechnen*.

Wir wählen eine natürliche Zahl $m \geq 2$, den sogenannten *Modulus*. Die grundlegendste Operation ist nun, den Rest einer ganzen Zahl bei der Division durch m zu nehmen. Die resultierende ganze Zahl, welche immer zwischen 0 und $m-1$ (einschließlich) liegt, heißt der *Rest von a modulo m* und wird mit $a \bmod m$ bezeichnet.

Beim Rechnen *modulo* m rechnet man in der Menge der natürlichen Zahlen $\{0, \ldots, m-1\}$ und nach jeder Rechenoperation wird der Rest modulo m genommen: Wenn a und b zwei natürliche Zahlen kleiner als m sind, ist das Resultat der *Moduloaddition* $(a + b) \bmod m$ und das Resultat der *Modulomultiplikation* $(a \cdot b) \bmod m$. Ferner ist für die Zahl a und eine natürliche Zahl e das Resultat der Moduloexponentiation $a^e \bmod m$.

Für eine Primzahl p und zwei natürliche Zahlen a, b zwischen 1 und $p-1$ (1 und $p-1$ einschließlich) ist auch $(a \cdot b) \bmod p \neq 0$. Dies kann als Analogon der Tatsache betrachtet werden, dass das Produkt von zwei Zahlen, die nicht 0 sind, niemals 0 ist. Man nennt die Menge $\{0, \ldots, p-1\}$ mit den gegebenen Rechenoperationen den *endlichen Primkörper* zur Primzahl p und bezeichnet ihn mit $\mathbb{F}_p$. Diese Rechenbereiche haben nun in vielerlei Hinsicht analoge Eigenschaften zum Bereich der rationalen Zahlen, $\mathbb{Q}$; sie können als endliche Analoga zum unendlich großen Bereich $\mathbb{Q}$ betrachtet werden.

Für das Verfahren von Diffie und Hellman ist relevant: Wenn p, a und e gegeben sind, kann $a^e \bmod p$ qualitativ schnell, d.h. in polynomieller Laufzeit, berechnet werden.

Das Verfahren ist nun wie folgt: Zunächst einigen sich Alice und Bob auf eine (große) Primzahl p und eine natürliche Zahl $a < p$. Die beiden Zahlen können jedermann bekannt sein, Alice und Bob können auch auf schon zuvor gewählte

Das Diffie-Hellman Protokoll

Alice und Bob einigen sich (in der Öffentlichkeit)
auf eine geeignete Primzahl p und
eine geeignete natürliche Zahl $a < p$.

Alice	**Bob**
Wählt $x < \mathrm{ord}(a)$.	Wählt $y < \mathrm{ord}(a)$.
Berechnet $X := a^x \bmod p$.	Berechnet $Y := a^y \bmod p$.
Sendet X an Bob.	Sendet Y an Alice.
$\xrightarrow{X}$	
$\xleftarrow{Y}$	
Berechnet $Y^x \bmod p$.	Berechnet $X^y \bmod p$.

Alice und Bob haben dieselbe Zahl berechnet, da
$Y^x \bmod p = a^{xy} \bmod p = X^y \bmod p$.

und standardisierte Zahlen zurückgreifen. Aus diesem Grund nennen wir p und a *öffentliche Parameter*. Alice wählt nun eine natürliche Zahl x und Bob wählt eine Zahl y, beide halten ihre Zahlen geheim. Alice berechnet nun $X := a^x \bmod p$ und schickt dies an Bob und dieser berechnet $Y := a^y \bmod p$ und schickt dies an Alice. Nun ist $X^y \bmod p = a^{xy} \bmod p = Y^x \bmod p$. Diese Zahl kann von beiden berechnet werden und soll das gemeinsame Geheimnis sein.

Wir stellen uns nun auf den Standpunkt eines LAUSCHERS, der das mutmaßliche Geheimnis berechnen will. Dieser empfängt p, a, X, Y und will $X^y \bmod p = Y^x \bmod p$ berechnen. Das Problem, eine solche Rechnung durchzuführen, heißt nun *Diffie-Hellman Problem*.

Ein möglicher Ansatz ist es, aus p, a, X eine Zahl x mit $a^x \bmod p = X$ zu berechnen. Denn dann kann der Lauscher auch leicht $Y^x \bmod p$ berechnen. Das genannte rechnerische Problem ist das sogenannte *diskrete Logarithmusproblem*. Das Wort „diskret“ wird hier im Sinne von „endlich“ oder „unzusammenhängend“ verwendet. Dass das Wort auch die Bedeutung „zurückhaltend“ oder „nicht in der Öffentlichkeit“ hat, die in der Kryptographie passend erscheint, ist Zufall.

Ein Jahr nach der Arbeit von Diffie und Hellman veröffentlichten Ron Rivest, Adi Shamir und Leonard Adleman ein Verfahren, bei dem jeder Benutzer ein Schlüsselpaar bestehend aus einem sogenannten *privaten* und einem *öffentlichen* Schlüssel hat ([28]). Die Idee dieses sogenannten *RSA-Verfahrens* ist wie folgt: Alice kann nun ihren öffentlichen Schlüssel allgemein verbreiten und jede Person kann diesen Schlüssel benutzen, um Alice eine verschlüsselte Nachricht zu senden. Aber nur Alice kann mittels ihres privaten Schlüssels die ihr gesandte Nachricht lesen. Da man nun zwei Schlüssel mit markant verschiedenen Rollen benutzt, spricht

man hier von einem *asymmetrischen Verschlüsselungsverfahren*, wohingegen die klassischen Verschlüsselungsverfahren mit einem einzigen, gemeinsamen Schlüssel *symmetrische Verschlüsselungsverfahren* genannt werden.

Auch dieses Verfahren beruht auf der Schwierigkeit eines rechnerischen Problems, in diesem Fall auf dem Problem, Produkte von zwei (großen) Primzahlen zu faktorisieren; dieses Problem wurde oben schon diskutiert. Nach heutigem Forschungsstand sind das diskrete Logarithmusproblem (bei zufälliger Wahl von p und a) und das genannte Faktorisierungsproblem bei gleicher Eingabelänge als etwa gleich schwer anzusehen; mehr zur Schwierigkeit des diskreten Logarithmusproblems in Abschnitt 3.2.6.

Neben der Verschlüsselung von Texten kann das RSA-Verfahren auch zum Signieren verwendet werden: Ein Text kann signiert werden, indem er mit dem privaten Schlüssel verschlüsselt wird. Eine mutmaßliche Signatur kann dann durch Entschlüsselung mittels des öffentlichen Schlüssels überprüft werden.

Für ihren Beitrag zur Kryptographie haben sowohl Rivet, Shamir und Adleman als auch Diffie und Hellman den Turing Award erhalten, den prestigereichsten Preis der Informatik, benannt nach einem der Pioniere der Informatik, Alan Turing: Rivest, Shamir und Adleman im Jahr 2002 und Diffie und Hellman im Jahr 2015.

3.2.5. Protokolle, aktive Angreifer und reduktive Sicherheitsbeweise. Nehmen wir an, Alice will mit Bob über das Internet vertraulich kommunizieren. Sie benutzt das Diffie-Hellman Verfahren mit konkreten Parametern p und a, um ein gemeinsames Geheimnis mit Bob zu vereinbaren und dann das Verschlüsselungsverfahren AES mit dem gemeinsamen Geheimnis als Schlüssel.

Wie soeben aufgezeigt, kann ein Lauscher das Diffie-Hellman Verfahren (für konkrete Parameter p und a) genau dann brechen, wenn er das sogenannte Diffie-Hellman Problem für dieselben Parameter lösen kann. Man könnte versucht sein, hieraus zu schließen, dass Alice ein adäquates Verfahren für ihr Ziel gewählt hat, wenn das Diffie-Hellman Problem für die konkreten Parameter unlösbar ist und kein Angriff auf AES bekannt ist.

Dies ist jedoch nicht der Fall. Der Grund hierfür ist einfach: Ab dem Moment des Schlüsselaustauschs könnte ihre Kommunikation zu Bob von einem *aktiven Angreifer* abgefangen werden, welcher sich als Bob ausgibt. Alice würde so vertrauliche Informationen an den Angreifer schicken.

Vielleicht würde Alice dies nach einer Zeit merken. Ein Angreifer kann aber auch noch geschickter vorgehen: Er gibt sich gegenüber Alice als Bob und gegenüber Bob als Alice aus und vereinbart mit beiden einen Schlüssel. Er kann dann jeweils die Nachrichten von Alice an Bob sowie von Bob an Alice entschlüsseln, lesen, wieder verschlüsseln und weiterleiten. Man spricht hier von einem sogenannten *man-in-the-middle Angriff*.

Die aufgezeigte Disfunktionanlität des Verfahrens kommt aufgrund vollständig fehlender Authentisierung zustande. Die aufgezeigten Probleme können aber auch abstrakter interpretiert werden: Ein kryptographisches Verfahren kann gegenüber einem AKTIVEN ANGREIFER UNSICHER sein, selbst wenn es gegenüber einem LAUSCHER, d.h. einem PASSIVEN ANGREIFER SICHER ist.

Während beim Diffie-Hellman Verfahren die Unsicherheit gegenüber aktiven Angreifern besonders evident ist, sind aktive Angriffe auch schon bei klassischen kryptographischen Verfahren relevant gewesen. So wurde eine Idee für einen aktiven Angriff auch schon in Abschnitt 2.2 beschrieben: Man lässt eine vorgegebene Nachricht vom Gegner verschlüsseln und versucht, das resultierende Paar von Klar- und Chiffriertext einzusetzen, um Informationen über Nachrichten zu erhalten, die man verschlüsselt erhält.

Kryptographische Verfahren, insbesondere interaktive kryptographische Verfahren, erlauben oftmals eine unübersichtliche Vielzahl von Manipulationsmöglichkeiten durch aktive Angreifer. Manche davon sind offensichtlich, andere wiederum nicht. Um von der Sicherheit eines Verfahrens (mit konkreten Parametern) überzeugt zu sein, hätte man gerne starke Argumente, dass das Verfahren sicher bleibt, egal welche Strategie ein Angreifer wählt.

Bei so gut wie jedem heutzutage verwendeten Verfahren ist es zumindest denkbar, dass die Sicherheit auch von passiven Angreifern kompromittiert werden kann, indem für ein bestimmtes zugrundeliegendes algorithmisches Problem ein neuer, überraschend effizienter Algorithmus gefunden wird. Beispielsweise beruht die Sicherheit des Diffie-Hellman Verfahrens mit konkreten Parametern gegenüber passiven Angreifern auf der Schwierigkeit des entsprechenden Diffie-Hellman Problems mit ebendiesen Parametern. Da solche grundlegenden Angriffe sowieso nicht verhindert werden können, versucht man, die Argumentation explizit auf der Schwierigkeit des unterliegenden Problems aufzubauen. Für eine gegebene Aufgabe ist nun das Ziel, effiziente Verfahren zu finden, für die eine große Klasse von Angriffen ausgeschlossen werden kann, wenn nur ein zugrundeliegendes algorithmisches Problem schwer ist.

Ein derartiges Vorgehen erfordert zunächst einmal adäquate mathematisch fundierte aber auch handhabbare Definitionen. Alleine schon die Aufgabe, solche zu finden, ist oftmals keine leichte.

Die Probleme beginnen schon beim Begriff des Verfahrens. Die gängige mathematische Definition beruht auf interagierenden Algorithmen, deren interne Rechnungen für den Angreifer unsichtbar sind. Man kann auch sagen, dass von den internen Rechnungen abstrahiert wird und nur die Ein-Ausgabe-Beziehungen betrachtet werden. Ein so abstrahiertes Verfahren heißt in der Kryptologie ein *Protokoll*. Ein Protokoll kann stets für beliebig große Eingaben verwendet werden und umfasst alle notwendigen Schritte. Ein Protokoll für das oben eher informell beschriebene Diffie-Hellman Verfahren beginnt beispielsweise mit einer Startup-Phase. In dieser Phase wird nach der Eingabe der Parametergröße (wie beispielsweise „1000 bit“) ein passendes Paar p, a gewählt wird. So ein Protokoll ist zu unterscheiden von einer *Implementierung*, die aus Computerprogrammen besteht, die die Ein-Ausgabe-Beschreibungen verwirklichen. Ein *Angreifer* ist dann stets ein Algorithmus, der mit dem Protokoll interagiert.

Eine noch größere Herausforderung ist es, adäquate formale Definitionen von „sicher“ für verschiedene Aspekte der Kryptographie wie Vereinbarung eines gemeinsamen Geheimnisses, Verschlüsselung oder Signatur zu finden.

Die gängigen Definitionen spiegeln hier den in Abschnitt 3.2.3 beschriebenen

komplexitätstheoretischen Gesichtspunkt wider. Dies bedeutet, dass keine absoluten Aussagen über die Sicherheit für konkrete Eingabelängen gemacht werden sondern QUALITATIVE Aussagen für beliebig große Eingabelängen. Der in Abschnitt 3.2.3 eingeführten Terminologie folgend betonen wir dies, indem wir den Ausdruck „qualitativ sicher“ benutzen.

Auf der Grundlage verschiedener Angriffsszenarien gibt es einige, verwandte formale Definitionen von *qualitativ sicher*. Üblicherweise basiert man die Definitionen heutzutage auf der Idee von *Spielen*. Man stellt sich hier einen „intelligenten“ Angreifer (auch *Gegner* genannt) und einen simplizistischen *Herausforderer* vor. Man sollte hierbei beachten, dass der Angreifer ein Algorithmus ist.

Um eine Idee dieses Ansatzes zu vermitteln, geben wir nun eine leicht informelle Beschreibung des stärksten gegenwärtig betrachteten Begriffs qualitativer Sicherheit für symmetrischer Verschlüsselungsverfahren, der *Ununterscheidbarkeit unter adaptiven Angriffen mit selbst gewählten Chiffriertexten* (*Indistiguishability under adaptive chosen cipher text attacks, IND-CCA2*). Der Angriff basiert auf dem folgenden Spiel.

- Nach Eingabe der Schlüssellänge wird von einem sogenannten Setup-Algorithmus ein geheimer Schlüssel (randomisiert) ausgewählt.
- Der Angreifer wählt einige Texte, sendet sie an den Herausforderer mit der Aufforderung, diese entweder zu ver- oder zu entschlüsseln. Der Herausforderer sendet die Ergebnisse an den Angreifer zurück.
- Der Angreifer wählt zwei verschiedene neue Texte M_1 und M_2. Er schickt beide an den Herausforderer. Dieser wählt einen derbeiden Texte mit gleicher Wahrscheinlichkeit aus, verschlüsselt sie zu C und sendet C zurück an den Angreifer.
- Der Angreifer wählt erneut einige Texte verschieden von C, die verschlüsselt und einige, die entschlüsselt werden sollen. Er sendet sie an den Herausforderer, der die gewünschten Operationen durchführt und die Ergebnisse zurücksendet.
- Der Angreifer entscheidet sich für M_1 oder M_2.

Während des ganz Spiels kann der Angreifer seine Strategie mittels zuvor erhaltener Information anpassen. Der Angreifer *gewinnt*, wenn er sich für denjenigen Text entscheidet, den der Herausforderer in Schritt 3 gewählt hat. Man beachte: Wenn der Angreifer nur rät, gewinnt er mit einer Wahrscheinlichkeit von $\frac{1}{2}$.

Wir nennen einen Angreifer *qualitativ erfolgreich*, wenn es qualitativ schnell ist und er mit einer Wahrscheinlichkeit gewinnt, welche qualitativ größer als $\frac{1}{2}$ ist. Ein Verfahren wird nun *IND-CCA2-sicher* genannt, wenn es keinen qualitativ erfolgreichen Angreifer für das soeben beschriebene Spiel gibt.

Auf das Spiel zurückkommend merken wir noch an, dass in Schritt 3 der Angreifer sogar M_1 oder M_2 (oder beide) als identisch zu einem in Schritt 2 gewählten Klartext wählen kann. Er kann auch, was gewisser Weise ähnlich ist, in Schritt 4

die Texte M_1 oder M_2 an den Herausforderer zur Entschlüsselung schicken. Die einzige Aufforderung des Angreifers an den Herausforderer, die nicht erlaubt ist, ist nach der Entschlüsselung von C zu fragen. Dies impliziert, dass ein Verschlüsselungsverfahren um IND-CCA2-sicher zu sein randomisiert sein muss.

Definitionen auf Spielen zu basieren war eine richtungsweisende Idee. Für diese Idee und verwandte Beiträge zu den mathematischen Grundlagen der Kryptographie erhielten Shafi Goldwasser und Silvio Micali im Jahre 2012 den Turing-Award.

Auf Grundlage eines Angriffsszenarios wie dem beschriebenen kann man dann versuchen, ein *reduktives Sicherheitsresultat* eines gegebenen Verfahrens zu beweisen. Hierfür fixiert man zusätzlich ein unterliegendes rechnerisches Problem. Ein reduktives Sicherheitsresultat, auch eine *Sicherheitsreduktion* oder einfach nur eine *Reduktion* genannt, ist dann eine mathematische Aussage der Form: Jeder qualitativ sichere Angriff der betrachteten Art auf das Protokoll führt zu einem qualitativ schnellen Algorithmus für das rechnerische Problem. Wenn so ein Resultat bewiesen ist, erhält man: Wenn es keinen qualitativ schnellen Algorithmus für das rechnerische Problem gibt, ist das Protokoll qualitativ sicher bezüglich der betrachteten Art von Angriffen.

Im Idealfall würde nun die Sicherheit (bezüglich einer präzisen und möglichst allgemeinen Definition) geeigneter Protokolle für die unüberschaubare Vielzahl kryptographischer Anwendungen der modernen Zeit mittels reduktiver Sicherheitsbeweise auf die Schwierigkeit einer kleinen Menge algorithmischer Probleme wie das Faktorisierungsproblem, das diskrete Logarithmusproblem oder das Diffie-Hellman Problem zurückgeführt. Diese Basisprobleme sollten dann eingehend von einer breiten wissenschaftlichen Gemeinde untersucht werden. Dieser rigorose Zugang wird besonders in einem zweibändigen Werk von Oded Goldreich von 2001 und 2004 Namens *Foundations of Cryptography* propagiert, welches als eine erste Konsolidierung des Gebiets gelten kann ([14]).

Man sollte jedoch beachten, dass es, wie schon in Abschnitt 3.2.3, immer eine Lücke zwischen komplexitätstheoretischen Überlegungen und Praxis gibt. Bei der praktischen Verwendung eines Protokolls gibt es, auch wenn ein reduktiver Sicherheitsbeweis bezüglich einer starken formalen Definition und auf Basis eines scheinbar starken unterliegenden rechnerischen Problems geführt wurde, sogar mehrere potentielle Probleme:

- Für ein praxisrelevantes Resultat muss bestimmt werden, für welche Eingabelänge (oder Parameter) das Protokoll verwendet werden soll. Wenn man den Ansatz mittels reduktiver Sicherheitsresultate ernst nimmt, muss man hierfür wie folgt vorgehen: Man wählt ein reduktives Sicherheitsresultat bezüglich eines bestimmten unterliegenden algorithmischen Problems. Das Resultat muss nicht nur qualitativ sondern explizit und quantitativ die rechnerische Komplexität von Angriffen und des rechnerischen Problems verbinden. Dann überlegt man sich, für welche Eingabegröße dieses algorithmische Problem praktisch unlösbar ist. Auf Grundlage dieser zwei Aussagen rechnet man aus, wie groß die Schlüssellänge sein sollte, damit kein praktisch relevanter Angriff möglich ist, wenn das unterliegende Problem tatsächlich so schwer wie angenommen ist.

Dies wird allerdings oft nicht gemacht, insbesondere, weil die Schlüssellänge dann unhandhabbar groß und die Verfahren zu langsam würden. Stattdessen werden oftmals kürzere Schlüssellängen gewählt oder ganz andere Verfahren betrachtet, die von den rigoros analysierten inspiriert, aber doch unterschiedlich sind.

- Selbstredend muss die Implementierung der Beschreibung entsprechen, darf also keinen Fehler enthalten. Dies auszuschließen ist schwierig.

- Auch auf Implementierungen, die der Spezifikation entsprechen, gibt es oftmals noch Angriffsmöglichkeiten. Auch wenn die Angriffsszenarien sehr weitgehend sind, wird immer angenommen, dass ein Angreifer nichts über die internen Rechnungen erfährt. Konkrete Produkte „strahlen“ aber oftmals ab, im wahrsten oder im übertragenen Sinne. Ein Beispiel sind die schon in Abschnitt 3.2.1 erwähnten Angriffe mittels der Rechenzeit.

3.2.6. Der Einfluss der Zahlentheorie. Vom mathematischen Standpunkt aus betrachtet fallen das diskrete Logarithmusproblem und das Faktorisierungsproblem nicht nur in den Bereich der Komplexitätstheorie, sondern auch in die Zahlentheorie. Mit der Arbeit von Diffie und Hellman wurde somit eine Beziehung von der Kryptologie zu diesem sehr etablierten Feld der reinen Mathematik hergestellt. Diese Beziehung ist besonders bemerkenswert, da Mathematiker noch vor einigen Jahrzehnten davon ausgingen, gerade die Zahlentheorie sei immun gegen Anwendungen, insbesondere im militärischen Bereich. So schreibt beispielsweise der berühmte Zahlentheoretiker Godfrey Harold Hardy in seinem Buch *A Mathematician's Apololgy* aus dem Kriegsjahr 1940, „echte“ (tiefsinnige) Mathematik sei „harmlos und unschuldig“ und konkret, es sei „sehr unwahrscheinlich“, dass irgendwer ein kriegerisches Ziel finden werde, dem mit Zahlentheorie gedient würde.

Überlegungen aus der Zahlentheorie und verwandten Gebieten der reinen Mathematik sind heutzutage aus der Kryptographie nicht mehr wegzudenken. Besonders deutlich wird dies beim diskrete Logarithmusproblem, also dem folgenden algorithmischen Problem:

Gegeben eine Primzahl p, eine natürliche Zahl $a < p$ und eine weitere natürliche Zahl $b < p$, für welche es ein x mit $a^x \bmod p = b$ gibt, berechne so ein x.

Der naheliegende erste Lösungsversuch ist, für gegebene p, a, b nacheinander für $x = 0, 1, 2, 3, \ldots$ zu testen, ob die gewünschte Gleichung $a^x \bmod p = b$ erfüllt ist.

Für die Analyse dieser einfachen Methode sind Überlegungen der elementaren Zahlentheorie relevant:

Es ist naheliegend, dass der Aufwand dieser Methode davon abhängt, wieviele Werte $a^x \bmod p$ annehmen kann. Diese Anzahl nennt man die *Ordnung* von a *modulo* p und bezeichnet sie mit $\mathrm{ord}(a)$. Die Werte $a^x \bmod p$ liegen zwischen 1 und $p-1$ (1 und $p-1$ eingeschlossen), also ist die Ordnung höchstens $p-1$. Ferner gilt, wie man beweisen kann: $a^{\mathrm{ord}(a)} \bmod p = 1$. Wenn man dies wieder fortgesetzt mit a multipliziert, erhält man $a^{\mathrm{ord}(a)+1} \bmod p = a$, $a^{\mathrm{ord}(a)+2} \bmod p = a^2 \bmod p$ und allgemein für jede natürliche Zahl e: $a^{\mathrm{ord}(a)+e} \bmod p = a^e \bmod p$. Es folgt, dass jeder mögliche Wert von $a^x \bmod p$ von genau einem x zwischen 0

und $\mathrm{ord}(a)-1$ (einschließlich) angenommen wird. Wenn man $x = 0, \ldots, \mathrm{ord}(a)-1$ laufen lässt, gibt es also genau ein solches x mit $a^x \bmod p = b$. Für festes p und a und vollkommen zufälliges b benötigt man im Durchschnitt $\frac{\mathrm{ord}(a)}{2}$ Versuche, bis man die Lösung gefunden hat.

Nun hat bereits Carl Friedrich Gauß in seinem berühmten 1801 erschienen Werk „Disquisitiones Arithmeticae“ bewiesen, dass für jede Primzahl p ein a mit Ordnung $p-1$ existiert. Betrachten wir so ein Paar p, a, d.h. sei p eine Primzahl und $a < p$ eine positive ganze Zahl von Ordnung $p-1$.

Mit dem bisherigen Ansatz mittels Ausprobieren braucht man nun ungefähr eine Zeit, die durch p gegeben ist. Zum Vergleich: Um für p, a und ein $x < p$ den Wert $a^x \bmod p$ zu berechnen, braucht man eine Zeit, die in etwa durch $\log_2(p)^3$ gegeben ist. Wenn p beispielsweise 100 bit (etwa 30 Stellen im Zehnersystem) hat, ist $\log_2(p)^3$ etwa 300, während p etwa 2^{100}, d.h. etwa 10^{30} ist. Der Unterschied ist gewaltig.

Schon zum Zeitpunkt der Veröffentlichung des Artikels von Diffie und Hellman war bekannt, dass man nicht nur eine Laufzeit von etwa p, sondern etwa $\sqrt{p}$ erreichen kann. Die Idee hierfür kann so beschrieben werden: Man berechnet Zahlen $a^c \bmod p$ und $a^d b \bmod p$ ab für zufällige $c, d < p-1$ und speichert die Ergebnisse ab. Man sucht dann nach einer sogenannten *Kollision* $a^c \bmod p = a^d b \bmod p$. So eine Kollision führt zu $a^{c-d} \bmod p = b$, falls $c \geq d$ und $a^{c-d+(p-1)} \bmod p = b$, falls $c < d$. Vielleicht überraschenderweise benötigt man im Durchschnitt nur etwa $\sqrt{p}$ Ergebnisse $a^c \bmod p$ und $a^d b \bmod p$, bevor man eine gewünschte Kollision finden kann.

Dies kann mittels einer klassischen zahlentheoretischen Methode, die unter dem Namen *Chinesischer Restsatz* bekannt ist, weiter verbessert werden. Insgesamt kann man eine Laufzeit erreichen, die in etwa durch $\sqrt{\ell}$ gegeben ist, wobei ℓ der größte Primteiler von $p-1$ ist.

Nun ist für eine Primzahl $p \geq 5$ die Zahl $p-1$ nie eine Primzahl, weil sie gerade und nicht 2 ist. Damit wird es interessant, Primzahlen p zu betrachten, für die $\frac{p-1}{2}$ auch eine Primzahl ist. Solche Primzahlen heißen *Sophie-Germain Primzahlen*. Interessanterweise ist nicht bewiesen, dass es unendlich viele solche Primzahlen gibt, dies wird aber vermutet.

Wenn wir Sophie-Germain Primzahlen betrachten, ist die Laufzeit der bisher betrachteten besten Methoden wiederum etwa $\sqrt{p}$.

Es gibt allerdings noch eine andere Methode, das diskrete Logarithmusproblem zu lösen, die als *Relationensuchmethode* bezeichnet werden kann. Diese Methode ist wesentlich effizienter als die Kollisionsmethode, wenn die Ordnung von a größenordnungsmäßig gleich p ist. Sie wurde schon 1922 vom Mathematiker Maurice Kraitchik in seinem Werk “Théorie des Nombres” ([22]) entwickelt, fiel dann aber in Vergessenheit und wurde nach der Veröffentlichung des Artikel von Diffie und Hellman wiederentdeckt. Diese Methode soll hier nicht dargestellt werden. Es sei nur angemerkt, dass die Relationensuchmethode benutzt, dass man viele natürliche Zahlen in Produkte von wesentlich kleineren (Prim-)Zahlen faktorisieren kann.

Es stellte sich nun die Frage, ob es eine Variante des beschriebenen diskreten Logarithmusproblems gibt, für die die genannten Algorithmen nicht funktionieren.

Hierzu möchte man den Rechenbereich $\{1, \ldots, p-1\}$ mit der Modulomultiplikation durch einen anderen passenden Rechenbereich ersetzen. Es stellt sich hierbei heraus, dass man die Kollisionsmethode unter keinen Umständen vermeiden kann. Der Grund ist, dass die Kollisitionsmethode in direkter Weise auf der Rechenoperation selbst beruht. Aber gibt es auch einen Rechenbereich, in dem keine bessere Methode bekannt ist, in welchem also insbesondere die Relationensuchmethode nicht funktioniert?

Um diese Idee zu verwirklichen, wurde 1985 unabhängig voneinander von Victor Miller und Neal Koblitz die *Kryptographie mit elliptischen Kurven* vorgeschlagen.

Elliptische Kurven sind keine Ellipsen, auch wenn der Name dies nahelegt; der Name beruht auf einem „historischen Zufall". Man kann allerdings ausgehend vom Kreis (der eine spezielle Ellipse ist) erklären, was elliptische Kurven sind: Im kartesischen Koordinatensystem ist der Einheitskreis, d.h. der Kreis mit Radius 1 und Mittelpunkt im Koordinatenursprung, durch die Gleichung $x^2 + y^2 = 1$ gegeben. Jeder Punkt $P = (x_P, y_P)$ auf dem Kreis lässt sich durch den Winkel α zur y-Achse beschreiben; es ist dann $x_P = \sin(\alpha), y_P = \cos(\alpha)$. Wenn nun so ein Punkt P und ein weiterer Punkt $Q = (\sin(\beta), \cos(\beta))$ gegeben ist, kann man die Winkel addieren und erhält so einen neuen Punkt $R = (\cos(\alpha + \beta), \sin(\alpha + \beta))$. Wir schreiben $P \star Q$ für diesen Punkt, wobei das Symbol „$\star$" willkürlich ist und durch ein beliebiges anderes Symbol ersetzt werden könnte.

Wir erhalten auf diese Weise eine Rechenoperation auf dem Kreis mit Radius 1, die die "Uhr-Operation" genannt werden könnte. Diese Operation erfüllt die üblichen Regeln der Assoziativität und Kommutativität, die von der Addition oder der Multiplikation reeller Zahlen bekannt sind. Darüberhinaus hat man mit $O := (0; 1)$ $P \star O = O \star P = P$ für jeden Punkt P; der Punkt O ist deshalb analog zur 0 für die Addition reeller Zahlen und zur 1 für die Multiplikation reeller Zahlen.

Nun benötigt man gar keine Winkel und trigonometrischen Funktionen für die Rechenoperation auf dem Kreis: Für Punkte $P = (x_P, y_P)$ und $Q = (x_Q, y_Q)$ sind die Koordinaten des resultierenden Punktes $R = P \star Q$ durch die rein algebraischen Formeln

$$x_R = x_P y_Q + x_Q y_P \quad \text{und} \quad y_R = y_P y_Q - x_P x_Q \tag{1}$$

gegeben. Wenn man nun eine negative Zahl d wählt, beschreibt die Gleichung

$$x^2 + y^2 = 1 + dx^2 y^2 \tag{2}$$

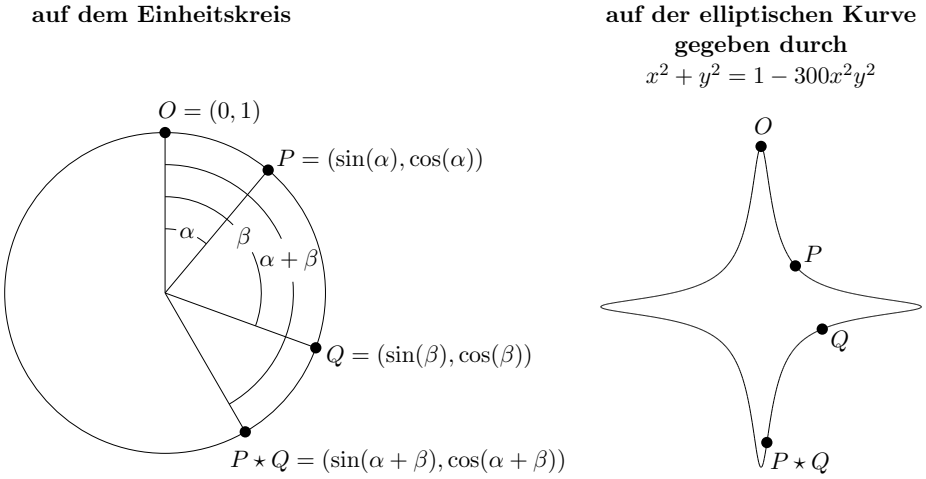
eine elliptische Kurve ([8], [2]). Interessanterweise kann man auch auf so einer Kurve eine Rechenoperation definieren. Für Punkte P und Q sind die Koordinaten des resultierenden Punktes $R = P \star Q$ durch

$$x_R = \frac{x_P y_Q + x_Q y_P}{1 + d x_P x_Q y_P y_Q} \quad \text{and} \quad y_R = \frac{y_P y_Q - x_P x_Q}{1 - d x_P x_Q y_P y_Q} \; . \tag{3}$$

gegeben. Diese Operation ist wiederum assoziativ und kommutativ und man hat wieder $O \star P = P \star O = P$ für jeden Punkt P.

Man beachte, dass man für $d = 0$ wieder den Kreis mit der Rechenoperation (1), der jedoch keine elliptische Kurve ist. Wir bemerken auch, da wir es sogleich benutzen werden, dass die Bedingung, dass d negativ sei, äquivalent zur Bedingung ist, dass d kein Quadrat einer anderen reellen Zahl sei.

Die Rechenoperationen



Nun werden in der Kryptographie nicht Lösungsmengen solcher Gleichungen über den rationalen oder den reellen Zahlen, sondern über endlichen Rechenbereichen betrachtet. Am häufigsten verwendet man hierbei die in Abschnitt 3.2.4 beschriebenen endlichen Primkörper $\mathbb{F}_p$, auf welche wir uns hier beschränken wollen.

Es sei also p eine Primzahl größer als 2 und $d \in \mathbb{F}_p$ mit $d \neq 0$. Wir betrachten die Lösungen der Gleichung (1) in $\mathbb{F}_p$, d.h. Tupel (x, y) mit $x, y \in \mathbb{F}_p$ und $(x^2 + y^2) \bmod p = (1 + dx^2y^2) \bmod p$. Um eine Rechenoperation auf der Lösungsmenge zu erhalten, muss man wieder sicherstellen, dass die Nenner in (3) stets nicht-Null sind. Hierfür benutzt man die Bedingung, dass d kein Quadrat eines anderen Elementes sei, welche nun auf die Rechnungen in $\mathbb{F}_p$ angepasst wird. Dies bedeutet dann, dass es kein $a \in \mathbb{F}_p$ mit $a^2 \bmod p = d$ gebe; es gibt genau $\frac{p-1}{2}$ solche Elemente in $\mathbb{F}_p$.

Der resultierende Rechenbereich wird üblicherweise mit $E(\mathbb{F}_p)$ bezeichnet. Die Idee ist nun, den Rechenbereich $\{1, \ldots, p-1\}$ mit der Modulomultiplikation durch solch einen Rechenbereich $E(\mathbb{F}_p)$ zu ersetzen. Dies ist in der Tat leicht möglich. Man kann dann wiederum von diskrete Logarithmen sprechen und erhält so das *elliptische Kurven diskrete Logarithmusproblem*. Entsprechend kann man auch Verfahren wie das von Diffie und Hellman auf elliptische Kurven anpassen.

Inzwischen wurde eine große Menge von weiteren Verfahren entwickelt, die auf Modulomultiplikation beruhen und die auf elliptische Kurven übertragen werden können. Hierbei ist unter anderem ein von Taher ElGamal entwickeltes Verfahren zu nennen, das als Alternative zu RSA benutzt werden kann.

Auch nach nun 30 Jahren Forschung ist die Kollisionsmethode für die meisten der hier betrachteten elliptischen Kurven immer noch die effizienteste Methode, um das elliptische Kurven diskrete Logarithmusproblem zu lösen. Dies bedeutet,

dass bei gleicher Schlüssellänge das Verfahren von Diffie und Hellman für elliptische Kurven ein wesentlich höheres Sicherheitsniveau ergibt als das ursprüngliche Verfahren. Gleiches gilt für andere kryptographische Protokolle, deren Sicherheit auf dem diskreten Logarithmusproblem beruhen.

So wird beispielsweise vom deutschen Bundesamt für Sicherheit in der Informationstechnik (BSI) empfohlen, bei Verwendung des diskreten Logarithmusproblems in elliptischen Kurven eine Schlüssellänge von mindestens 250 bit zu verwenden, während beim klassischen diskreten Logarithmusproblem und beim RSA-Verfahren eine Schlüssellänge von mindestens 3000 bit empfohlen wird ([4]).

Einem Leser, der sich für Kryptographie mit öffentlichen Schlüsseln und ihre Zahlentheoretischen Grundlagen interessiert, sei das Buch „Mathematics of Pulbic Key Cryptography“ von Steven Galbraith ([10]) empfohlen.

3.2.7. Smartcards. Bei der bisherigen Darstellung der Kryptographie mit öffentlichen Schlüsseln wurde ein fundamentales technisches Problem ausgeblendet: Wie kann der private Schlüssel vor unerwünschten Zugriffen geschützt werden? Nun kann man den Schlüssel auf viele Weisen speichern, in den 1980er Jahren beispielsweise auf einer Diskette. Um den Schlüssel zu benutzen, muss er allerdings zuerst in den Speicher eines Computers geladen werden. Dies stellt ein Sicherheitsrisiko dar, insbesondere wenn der Computer mit anderen Computern verbunden ist, wovon man heutzutage ausgehen muss.

Eine naheliegende Lösung ist, ein kleines Gerät zu schaffen, das seinen eigenen kleinen Computer mit Speicher und Mikroprozessor enthält. Hierdurch können kryptographische Anwendungen wie Authentisierung und digitale Signatur verwirklicht werden, bei denen der private Schlüssel das Gerät niemals verlässt.

Anfang der 1980 Jahre war die Miniaturisierung so weit fortgeschritten, dass die Idee mit sehr dünnen und etwa ein viertel Quadratzentimeter großen Miniaturcomputern verwirklicht werden konnte. Auf Plastikkarten in Kreditkartengröße aufgetragen erhält man so die sogenannten *Smartcards*.

Das Geheimnis, der private Schlüssel, soll nicht nur im normalen Betriebsmodus niemals die Karte verlassen, sondern es soll möglichst unter allen Umständen unmöglich sein, nutzbare Informationen über die internen Rechnungen zu erhalten, auch dann, wenn ein Angreifer die Umgebung der Karte vollkommen kontrolliert. Dies bedeutet beispielsweise, dass die Karte keine internen Informationen über Strahlung oder über Stromverbrauch abgeben darf.

Wenn die Karte in die Hände eines Gegners fällt, soll sie vollkommen nutzlos sein. Hierzu wird sie mittels eines Passworts geschützt, wobei die Karte sich selbst bei wiederholter Fehleingabe permanent sperrt. Hiernach soll es auch bei physikalischer Manipulation oder partieller Zerstörung der Karte unmöglich sein, die Karte zu benutzen oder Informationen über den Schlüssel zu erhalten, der ja auf der Karte gespeichert ist.

Die Vielzahl an potentiellen physikalischen Angriffen stellt für Kartenhersteller eine Herausforderung dar, die diese aber gut im Griff zu haben scheinen.

4. Aktuelle Entwicklungen und Herausforderungen

4.1. Allgegenwärtige Anwendungen und Vernetzung. Während Kryptographie zur Datensicherheit von Firmen und Regierungen schon seit den 1970er Jahren eingesetzt wird, sind seit dem Siegeszug des World Wide Web kryptographische Verfahren auch aus dem Alltagsgebrauch vieler, in den Industrieländern wohl der meisten, Menschen nicht mehr wegzudenken.

Automatisch wird Kryptographie mit öffentlichen Schlüsseln eingesetzt, wenn man eine Seite mit einer Adresse der Form `https://...` aufruft, was wiederum regelmäßig der Fall ist, wenn man eine Zahlung tätigt. Hier wird der Benutzer über die Verwendung von Kryptographie informiert, an anderer Stelle geschieht dies „transparent“ für den Benutzer, d.h. ohne dass dieser es überhaupt merkt.

Allein rund uns Auto kommt Kryptographie mehrfach zum Einsatz. Eine schon klassische Anwendung sind elektronische Schlüssel und Wegfahrsperren. Aber auch bei den Systemen für Straßenbenutzungsgebühren, wie beispielsweise beim deutschen Toll Collect, wird Kryptographie eingesetzt. Seit 2006 werden Manipulation des in der Europäischen Union für Busse und LKWs vorgeschrieben digitalen Tachographen mit Kryptographie verhindert. Ganz neue Herausforderungen wird die Kommunikation von Auto zu Auto stellen, besonders wenn dies Einfluss auf das automatische Verhalten von Autos hat. Zusätzlich zur Verwendung von Kryptographie in Anwendungen wird Kryptographie benutzt, um Software in elektronischen Geräten vor Manipulationen zu schützen. Mit der steigenden Bedeutung elektronischer Geräte in sicherheitsrelevanten Systemen wie Bremsen oder dem Steuerungssystem wird dieser Schutz immer relevanter.

Es sind drei miteinander verbundene Trends erkennbar: Mehr und mehr Geräte und Produkte werden mit Mikroprozessoren ausgestattet, diese Geräte werden immer stärker miteinander vernetzt und sie greifen dabei immer stärker auf gemeinsame außerhalb der Geräte selbst liegende Datenquellen zu.

Sowohl im geschäftlichen wie im privaten Bereich wird es immer üblicher, Daten nicht lokal, sondern in einer Rechnerwolke zu speichern und Dienste wie Dropbox, google drive oder icloud zu verwenden. Bei allen diesen Diensten wird Kryptographie eingesetzt, es bleibt aber die Frage, inwieweit die Daten wirklich sicher gegenüber Zugriff durch Mitarbeiter oder Behörden sind. Wenn man allerdings die Daten schon vor dem Abschicken verschlüsselt, wird dieses Problem vermieden. Es bleibt dann jedoch die Frage, ob lokal installierte Software auch genau das tut, was sie angibt oder nicht noch ein paar mehr Daten versendet.

Auch im häuslichen Bereich sind durch die Vernetzung kryptographische Herausforderungen entstanden oder werden in naher Zukunft entstehen. So wollen viele Staaten der Europäischen Union bis zum Jahr 2020 erreichen, dass eine Großzahl der Haushalte mit „intelligenten Zählern“ (Smart Meter) ausgestattet wird. Die hierbei auftretenden Datenschutzprobleme sollen mit Kryptographie gelöst werden. In Deutschland werden hierfür seit 2009 Pilotprojekte durchgeführt. Im Kontext der im Zuge der „Energiewende“ propagierten „intelligenten Netze“ (Smart Grids) mit automatischem, vom Netzverwalter aus gesteuerten, An- und Abschalten elektrischer Geräte werden noch einige kryptographische Herausforderungen auf uns zukommen.

Auch ganze Industrieanlagen und kritische Infrastruktur wie Pipelines, Stromnetze und Kraftwerke werden vernetzt und mit dem weltweiten Internet verbunden, was Anlass zu allen möglichen Horrorvisionen gibt. Eine radikale Lösung gegen Bedrohungen ist selbstredend, die Steuerung der Anlagen physikalisch vollständig von der Kommunikation nach außen zu trennen. Oftmals will man allerdings aus Sicherheitsgründen den Zugriff auf die Anlagen von außen vermeiden, während man gleichzeitig ein externes Monitoring ermöglichen will. Da alleine schon das Vorhandenseins eines von außen zugänglichen Kanals als Schwachstelle aufgefasst werden kann, bietet sich hier als Lösung an, einen Kanal zu schaffen, der aufgrund seiner physischen Realisierung keinen Datentransfer nach innen zulässt. Die Firma Waterfall Securities Solutions bietet Produkte an, die dieses Prinzip mittels eines Lasers verwirklichen. Mittels dieser Technik kann ein System geschaffen werden, welches ein wesentlich höheres Sicherheitsniveau bietet, als ein elektronisches Gerät, das auf einem kryptographischen Verfahren beruht.

4.2. Die Zukunft der Computer. Die Rechnenleistung von Computern in Bezug auf die aufzuwendenden finanziellen Mittel hat seit dem II. Weltkrieg ohne Unterbrechung rasant zugenommen. Durch diese Entwicklung sind immer wieder zuvor als sicher betrachtete kryptographische Systeme angreifbar geworden. Was wird hier die Zukunft bringen und welche Auswirkungen wird dies auf die Kryptologie haben?

Seit Beginn der 1970er Jahre ist die Integrationsdichte von Mikroprozessoren exponentiell angewachsen. Sie hat sich etwa alle zwei Jahre verdoppelt, eine Tatsache, die als *Moore'sches Gesetz* bekannt ist. Diese Erhöhung der Informationsdichte kann aufgrund klarer physikalischer Grenzen allerdings nicht beliebig weitergehen. Konkret wird zur Zeit in den modernsten Fabriken die sogenannte 14 Nanometer Technologie eingesetzt, was schon sehr bemerkenswert ist, wenn man bedenkt, dass ein Siliziumatom einen Radius von etwa 0,1 Nanometer hat. Es gibt Pläne für eine 5 Nanometer Technik für etwa das Jahr 2020, bei der man aber schon mit physikalischen Grenzen kämpfen muss. So kommt unterhalb von 7 Nanometer der Effekt der Quantentunnelung zum Tragen, das heißt, dass Elektronen die logischen Gatter durchdringen können. Aufgrund dieser Tatsachen kann man erwarten, dass die bisherige exponentielle Entwicklung relativ bald ausklingen wird.

Es ist aber denkbar, dass die Entwicklung dazu führt, dass in ganz neuer Weise quantenmechanische Phänomene für Computer nutzbar gemacht werden können. Die Quantenwelt ist eine für Menschen sehr seltsame Welt. Sie übersteigt immer wieder das menschliche Vorstellungsvermögen, was man insbesondere an den verschiedenen Interpretationen der Quantenmechanik sehen kann.

Schon im Jahr 1981 formulierte der Physiker Richard Feynman die Idee eines „Quantencomputers“ und im Jahr 1994 veröffentlichte der Mathematiker Peter Shor eine Skizze für potentielle Quantencomputer für das Faktorisierungs- sowie das für diskrete Logarithmenproblem. In einem mathematischen Modell, in welchem analog zu klassischen Rechenmodellen Quantencomputer idealisiert beschrieben werden, können diese Probleme mit Shors Ansatz in polynomiellem Aufwand, d.h. qualitativ effizient gelöst werden können. Für dieses richtungsweisende Resultat

erhielt Shor den prestigereichen Rolf Nevanlinna Preis für mathematische Aspekte der Informationswissenschaften der Internationalen Mathematischen Union im Jahr 1998.

Für die Berechnung diskreter Logarithmen werden hierbei, wie bei der in Abschnitt 3.2.6 beschriebenen Kollisionsmethode, nur die Rechenoperation selbst benutzt. Die Methode ist deshalb nicht nur auf das klassische diskrete Logarithmusproblem modulo einer Primzahl sondern auch auf das Problem in elliptischen Kurven anwendbar. Interessanterweise könnte der gegenwärtige wesentliche Vorteil von Systemen, die auf elliptischen Kurven beruhen, die relativ kurze Schlüssellänge, solche Systeme besonders verwundbar gegen Quantencomputer machen.

Es ist im Moment allerdings unklar, ob jemals so ein Quantencomputer verwirklicht werden wird, der mit klassischen Computern mithalten kann. Die bisherigen Versuche sind jedenfalls etwas ernüchternd: Im Jahr 2001 wurde mittels eines Quantencomputers basierend auf Shors Ideen die Zahl 15 faktorisiert, der gegenwärtige Rekord aus dem Jahr 2012 ist die Zahl 21.[1]

In den potentiellen Quantencomputern wie von Shor ersonnen werden Zustände analog zu klassischen Computern mit der Zeit manipuliert, was sehr schwer zu realisieren ist. Es gibt noch eine andere, eher passive Methode um Quantenphänomene nutzbar zu machen. Mit dieser Methode können die Ideen von Shor nicht verwirklicht werden, es ist aber denkbar, dass sie für einige Anwendungen zu überraschend effizienten Computern führt. Die Firma D-Wave entwickelt Computer, die auf dieser passiven Methode basieren. Die Computer scheinen wie geplant zu funktionieren, eine Prognose über die Leistungsfähigkeit der Technik scheint im Moment aber schwer möglich zu sein.

Interessanterweise scheint die amerikanische National Security Agency (NSA) davon auszugehen, dass in der Tat in den nächsten Jahrzehnten Quantencomputer im Sinne Shors realisiert werden können. So hat die NSA am 19. August 2015 bekannt gegeben, bei den öffentlich empfohlenen Verfahren, die auch Verfahren zur Sicherung von Geheimnissen der US-Regierung einschließen, „in der nicht zu weit entfernten Zukunft“ „den Übergang“ zu „quantenresistenten Verfahren“ „einzuleiten“ ([26]).

4.3. Krypto-Währungen und Krypto-Verträge. Die Krypto-Währung *bitcoin* ist in aller Munde. Die Kursentwicklung in Euro ist beeindruckend: 10.000 bitcoin bot ein Programmierer im Jahr 2010 für zwei Pizzen, die prompte Lieferung führte zur ersten Bezahlung in bitcoin überhaupt. Zum bisherigen Höchstkurs von 905 Euro pro bitcoin im Dezember 2013 ergibt sich, natürlich rein rechnerisch, ein stolzer Preis von etwa 9 Millionen Euro, auch zum Kurs von etwa 400 Euro pro bitcoin Ende 2015 sind die Pizzen doch noch recht teuer gewesen.

Obwohl die Entwicklung beeindruckend ist, gibt es im Moment keine Anzeichen dafür, dass bitcoin im tatsächlichen alltäglichen Gebrauch Relevanz erlangen könnte.

Das bitcoin Bezahlsystem ist zusammen mit der Anonymisierungssoftware TOR

[1] Es handelt sich nicht um Schreibfehler; es geht wirklich um die Zahlen 15 und 21, nicht etwa um Zahlen mit 15 bzw. 21 bits.

jedoch eine der entscheidenden technischen Systeme für anonymisierte Handelsplattformen im Internet. Der bekannteste dieser Märkte war die von 2011 bis 2013 betriebene Silk Road. Als diese Seite im Jahr 2013 vom FBI geschlossen wurde, gab diese an, in etwa einer Million Transaktionen sei ein Umsatz von über 9 Millionen bitcoin erzielt worden. Der Physiker Ross Ulbricht wurde als der unter dem Pseudonym Dread Pirate Roberts auftretende Betreiber der Seite identifiziert und wegen Drogenhandels und anderer Verbrechen zu lebenslanger Strafe ohne Aussicht auf Freilassung verurteilt. Ulbricht gab im Verfahren an, die Seite aus Idealismus gegründet zu haben, „um Menschen to befähigen, um in ihren Leben Wahlen zu treffen, für sich selbst in Privatheit und Anonymität“ und später nicht mehr beteiligt gewesen zu sein. Geglaubt hat man ihm dies nicht.

Trotz des drastischen Urteils liegt es nahe, dass Silk Road nicht der letzte erfolgreiche anonyme Handelsplatz gewesen sein wird, aus welcher Motivation heraus er auch immer betrieben werden wird. Eine besonders interessante Situation könnte entstehen, wenn ein Staat den Betrieb einer solchen – weltweit erreichbaren Plattform – für legal erklärt. Aufgrund eines in Aussicht stehenden Umsatzes in Milliarden Euro besteht besonders für kleine, ärmere Staaten ein Anreiz, dies zu tun, wenn nur der Gewinn versteuert wird.

Auch wenn anscheinend bitcoin nicht zu einem großen Sprung ansetzt, könnte dies für ein Teil des bitcoin-Protokolls der Fall sein: der *Blockketten-Technik*.

Oberflächlich werden bei bitcoin Rechte an „elektronischen Münzen“ übertragen. Eine erste Idee wäre, die Rechte an den Münzen in einem einzigen Journal zu verwalten. Der Verwalter des Journals wäre dann eine Art Aufbewahrungsbank. Bei Bitcoin wird jedoch ein anspruchsvolleres dezentrales Verfahren umgesetzt.

Dies geschieht mit Kryptographie mit öffentlichen Schlüsseln auf Basis von elliptischen Kurven. In etwa funktioniert es so: Wenn Alice eine Münze an Bob übertragen will, signiert sie mit ihrem privaten Schlüssel eine Verbindung (Konkatenation) aus der Münze und Bobs öffentlichen Schlüssel. Es muss nun aber sichergestellt werden, dass Alice tatsächlich berechtigt ist, die Münze zu übertragen. Hierzu werden (im Wesentlichen) alle Transaktionen von Anbeginn von bitcoin an gespeichert. Diese Lösung mag auf den ersten Blick der gewünschten Anonymität von bitcoin widersprechen. Diese wird dadurch sichergestellt, dass „Alice“, „Bob“ und so weiter nur virtuelle Konstrukte sind, die immer wieder neu erzeugt werden und hinter denen beliebige Personen stehen können.

Das Journal der Transaktionen wird nun nicht nur einmal sondern auf vielen verschiedenen sogenannten Knoten gespeichert. Genauer werden neue Transaktionen in Blöcken zusammengefasst und alle 10 Minuten wird in allen Knoten derselbe neue Block an das gespeicherte Journal angehängt. Dieses mehrfach gespeicherte Journal heißt *Blockkette (blockchain)*.

Indem man das bitcoin Protokoll leicht verändert, kann man Systeme zur dezentralen Verwaltung und Übertragung von Rechten verschiedener Art aufbauen. Eine naheliegende Möglichkeit ist es, so ein System zur Verwaltung von Wertpapieren zu benutzen.

Wertpapiere werden von Wertpapiersammelbanken verwaltet und von der Transaktion bis zum sogenannten „Settlement“ (der Eigentumsübertragung) vergehen in

der Regel zwei Tage, während deren die Vertragspartner das Risiko des Ausfalls der Gegenseite haben. Mit einem Verfahren wie blockchain würden die Wertpapiersammelbanken wegfallen und das Settlement wenige Minuten nach dem Deal stattfinden. Dies wäre sowohl effizienter als auch risikoarmer.

Im Moment gibt es einen regelrechten „Hype“ um blockchain und es sieht so aus, als ob diese Technik im Gegensatz zum anarchischen bitcoin wirklich die Finanzwelt verändern wird.

4.4. Edward Snowden und die NSA. Die „Edward Snowden Story“ wird in der Öffentlichkeit in der Regel als spannender Realkrimi wahrgenommen. Die Enthüllungen selbst haben zu einem undifferenzierten Gefühl von „die hören doch alles mit“ geführt.

In der Tat ist durch die Enthüllungen eine erhebliche Sammelwut zu Tage getreten. Andererseits: Dass ein für die Überwachung und Auswertung elektronischer Kommunikation zuständiger Geheimdienst – wohl im Rahmen der gesetzlichen Vorgaben – eben diese Kommunikation überwacht und mittels Filtertechniken auswertet, sollte schon zuvor klar gewesen sein. Noch klarer sollte gewesen sein, dass diese Technik auch dazu benutzt wird, Zielpersonen zu überwachen, besonders nachdem das entsprechende Land einen massiven Terrorangriff erlitten hat.

Auch sollte es allgemein bekannt gewesen sein, dass das Verschicken einer email dem Verschicken einer Postkarte gleicht, die jeder mitlesen kann, der sie in die Hand bekommt.

Wesentlich interessanter ist die Frage nach den Fähigkeiten der NSA bei verschlüsselter Kommunikation. Hierzu wurden jahrzehntelang viele Spekulationen angestellt. Die nun öffentlichen Dokumente erlauben zum ersten Mal einen Einblick in das Vorgehen und die Fähigkeiten der NSA auf diesem Gebiet.

Laut der Dokumente betreibt die NSA ein großes auf Kryptoanalyse angelegtes Programm namens *Bullrun*. In einer Präsentation des britischen Partnerdienstes GCHQ ([12]) heißt es, dieses Programm würde die „Fähigkeit, Verschlüsselung in spezifischen Netzwerkkommunikationen zu schlagen abdecken“ und eine „Vielzahl von extrem sensitiven Quellen und Methoden beinhalten“. Es offeriere „groundbreaking capabilites“, sei „extrem fragil“ und die Adressaten sollten weder nach den Quellen und Methoden, auf denen der Erfolg des Programms beruht, fragen, noch über sie spekulieren.

Die weiteren Dokumente und bekannte Tatsachen laden allerdings genau zu solchem Spekulieren ein, so dass wir dies nun auch tun werden.

4.4.1. Proaktives Vorgehen. Die Bemühungen der NSA setzen nicht erst bei schon etablierten Verfahren an. Vielmehr wird versucht, langfristig die Entwicklung in eine für den Dienst günstige Richtung zu lenken.

Nach einem der enthüllten Dokumente ([27]) standen in den Jahren 2011 bis 2013 jeweils zwischen 250 und 300 Millionen US Dollar für das „SIGINT Enabling Project“ zur Verfügung. Nach der Projektbeschreibung „nimmt das [...] Projekt die US und ausländische IT Industrie aktiv in Anspruch, um im Verborgenen Einfluss zu nehmen und/oder offen das Design ihrer kommerziellen Produkte zu beein-

flussen. Diese Änderungen machen das fragliche System mit dem Vorwissen der Veränderung nutzbar durch SIGINT Erfassung. Für den Konsumenten und andere Gegner jedoch bleibt die Systemsicherheit intakt.“[2] Die Projektmittel sollen unter anderem eingesetzt werden, um „Schwachstellen“ in Systeme „einzusetzen“ und um „Policies, Standards und Spezifikationen für kommerzielle Techniken der Kryptographie mit öffentlichen Schlüsseln zu beeinflussen“.

Letzteres ist augenscheinlich mindestens einmal gelungen, nämlich beim sogenannten *Dual-EC-Deterministic Random Bit Generator*, kurz *Dual_EC_DRBG*.

Kryptographische Protokolle benötigen oftmals „Zufall“ und dieser muss im Computer erzeugt werden. Nun kann man im Computer physikalisch „echten Zufall“ erzeugen, dies ist jedoch (pro bit) relativ zeitaufwendig. Deshalb verwendet man einen sogenannten *Pseudozufallsgenerator* oder *determinstischen Zufallsbitgenerator*, um aus einer echt zufälligen bit-Kette eine deutlich längere bit-Kette zu erzeugen. Die wesentlichen Anforderungen an so einen Generator sind, dass dieser sehr schnell ist und dass sich die Ausgabe nur unter unrealistisch hohem Rechenaufwand von einer echt zufälligen bit-Kette unterscheiden lässt.

Nun veröffentlichte im Jahr 2006 die US-amerikanische Standardisierungsbehörde NIST eine „Empfehlung für Zufallszahlerzeugung mit deterministischen Zufallsbitgeneratoren“. Wie schon in Abschnitt 3.2.1 erwähnt, gelten alle Standards oder auch Empfehlungen von NIST formal nur für die amerikanische Regierung und ihre Vertragspartner, werden oftmals aber de facto Industriestandards.

Eine Art dieser Generatoren hat den Namen Dual_EC_DRBG, wobei DRBG eine Abkürzung für „Deterministic Random Bit Generator“ ist. Zu dieser Art von Generatoren ist auch eine genaue Spezifikation mit konkreten zu verwendenden Parametern in dem Dokument enthalten. Diese Spezifikation kann nun als der erste Standard mit geheimer – aber dann doch offensichtlicher – Hintertür gelten.

Schon zwei Jahre vor Veröffentlichung des Dokuments von NIST wurde der so spezifizierte Generator von der Firma RSA Security als „default option“ in der viel benutzten kryptographischen Softwarebibliothek BSAFE implementiert und erst nach den Enthüllungen von Edward Snowden wieder entfernt.

Das „Dual_EC“ steht für „dual elliptic curve“ und in der Tat sind elliptische Kurven für diese Art von Generatoren von besonderer Relevanz. Das Konstruktionsprinzip ist dabei im Vergleich zu anderen Generatoren besonders klar und mathematisch elegant. Die Art der Generatoren ist eine effizientere Variante der am besten bekannten und untersuchten Generatoren der komplexitätsorientierten Kryptographie. Für geeignete Wahlen von Parametern kann die qualitative Sicherheit der Generatoren auf ein dem Diffie-Hellman Problem ähnliches Problem reduziert werden.

In der Praxis jedoch haben diese Generatoren den Nachteil, wesentlich langsamer als konkurrierende Generatoren zu sein. Außerdem hat, wie bereits im Jahr 2005 klar wurde, der konkrete von NIST spezifizierte Generator hat den weiteren

[2] “[The] project actively engages the US and foreign IT industries to covertly influence and/or overtly leverage their commercial products' designs. These design changes make the systems in question exploitable through SIGINT collection [...] with foreknowledge of the modification. To the consumer and other adversaries, however, the systems' security remains intact.”

Nachteil, dass die Ausgabe von einer echt zufälligen bit-Kette unterschieden werden kann, auch wenn die Abweichung klein ist ([13]). Offensichtlich wurden vom Gesichtspunkt der Sicherheit aus die Parameter absichtlich falsch gewählt.

Wie in [3] aufgezeigt, hatte die Wahl der Parameter ihren Grund: Der Generator kann immer noch als recht sicher für den Benutzer gelten und er hat gleichzeitig eine offensichtliche Hintertür für denjenigen, der die Parameter gewählt hat. Dies korrespondiert genau zu den im oben zu „Bullrun“ zitierten Dokument angegebenen Zielen. Da die Parameter für den von NIST spezifizierten Generator offiziell von der NSA berechnet wurden, kann man nur annehmen, dass die NSA eine Hintertür zu diesem Generator hat.

Wie konnte der NSA die Meisterleistung gelingen, dass ein Verfahren mit offensichtlicher Hintertür, das darüber hinaus auch noch langsam ist und auch neben der Hintertür eine Sicherheitsschwäche hat, in eine weit benutzte kryptographische Softwarebibliothek als Standardoption implementiert wurde?

Auf diese Frage ist selbstredend keine abschließende Antwort möglich, es gibt aber einige interessante Informationen. Laut Reuters zeigen von Snowden enthüllte Dokumente, dass die Firma RSA Security von der NSA 10 Millionen Dollar erhalten hat, um den Standard als „default option“ zu implementieren ([23]). RSA Securities antwortete, dass man „kategorisch“ abstreite, einen „‚geheimen‘ Vertrag mit NSA eingegangen zu sein, um einen bekanntermaßen fehlerhaften Zufallszahlengenerator in die BSAFE Verschlüsselungsbibliotheken einzufügen.“ RSA Security bemerkt, dass sie in der Tat mit der NSA zusammengearbeitet habe, die jedoch eine „ zuverlässige Rolle beim Bemühen der ‚community‘“ innegehabt habe, „Verschlüsselung zu stärken und nicht zu schwächen“. ([29]).

Die NSA ist in der Tat keine monolithische Organisation mit dem Ziel der Informationsbeschaffung und -analyse sondern hat auch einen defensiven Arm, das Information Assurance Directorate. Es gibt also einen internen Konflikt in der Organisation selbst. Dieser Konflikt wurde auch von einer „Review Group on Intelligence and Commuincations Technologies“ angesprochen, die vom amerikanischen Präsidenten Barack Obama nach den Enthüllungen von Edward Snowden eingesetzt worden war ([6]). Das Komitee stellte fest, dass „die NSA nun mannigfaltige Aufträge und Mandate habe, einige von ihnen unscharf, inhärent widersprüchlich oder beides“ und empfahl, das Information Assurance Directorate vom Rest der NSA abzuspalten. Ironischerweise wurde jedoch nicht nur der Vorschlag nicht umgesetzt, sondern die NSA verkündete im Februar 2016 eine Reorganisation, genannt NSA21 mit umgekehrter Zielrichtung: Die Information Assurance und Signals Intelligence Direktorate sollen in einem einzigen Direktorat namens Operations zusammengefasst werden, in welcher Mitarbeiter offiziell „auf beiden Seiten“ arbeiten sollen ([25]). Es ist nicht schwer vorherzusagen, dass dieser Zusammenschluss nicht hilfreich für das Wiedererlangen von Vertrauen seitens der Industrie sein wird.

In Deutschland ist man sich übrigens den Zielkonflikten schon lange bewusst. So wurden die defensiven kryptographischen Aktivitäten schon 1991 aus dem Bundesnachrichtendienst herausgenommen und um sie herum eine neue Behörde, das BSI aufgebaut.

4.4.2. Kryptoanalyse. Welche „klassische“ Kryptoanalyse die NSA genau betreibt, geht aus den Dokumenten nicht hervor. Es wäre auch ein besonderer Zusammenbruch von Geheimhaltung, wenn der externe Mitarbeiter Edward Snowden hierzu Dokumente in die Hand bekommen hätte. Anwendungen der Kryptoanalyse und einen allgemeinen Eindruck vom Aufbau der Systeme kann man allerdings den Dokumenten entnehmen.

Virtuelle Private Netzwerke sind sichere Verbindungen über das Internet, die beispielsweise bei externen Zugängen in Firmennetzwerke zum Einsatz kommen. Oftmals kommt dabei das sogenannte *IPsec Protokoll* zum Einsatz. Nach den Dokumenten hat die NSA ein System zum Analysieren von Daten solcher Verbindungen aufgebaut. Wenn auch nicht bekannt ist jedoch, wie dieses System funktioniert, wird in dem Artikel [1] das Folgende glaubhaft argumentiert:

Die meisten Verbindungen wurden vor Erscheinen des Artikels mit einem Diffie-Hellman Schlüsselaustausch modulo einer Primzahl p von 1024 bit begonnen. Nun haben schon im Jahr 2005 Forscher Möglichkeiten aufgezeigt, wie man potentiell mit einem damaligen Einsatz von etwa einer Milliarde Euro eine Maschine bauen könnte, die innerhalb eines Jahres einen entsprechenden diskreten Logarithmus berechnen könnte und somit solch einen Schlüsselaustausch knacken könnte ([9]).[3]

Eine solche Maschine wäre allerdings per se recht sinnlos für einen Geheimdienst; man will ja nicht nur einen Schlüsselaustausch pro Jahr knacken. Das Problem, viele diskrete Logarithmen zu berechnen, ist in der Praxis allerdings oftmals wesentlich einfacher, als man erwarten könnte: Nach Informationen aus [1] werden ungefähr zwei Drittel aller Schlüsselaustausche in VPNs bezüglich derselben Primzahl p durchgeführt. Dies bedeutet, dass man immer wieder diskrete Logarithmen modulo nur einer einzigen Primzahl berechnen muss. Wenn es nun auch sehr aufwendig ist, die Berechnung von diskreten Logarithmen modulo einer festen Primzahl in Gang zu bringen, so ist es doch nach der Startphase überraschend einfach, beliebig viele diskrete Logarithmen auszurechnen.

Aufgrund des Preisverfalls für Rechenleistung könnte die NSA mit Kosten von einigen hundert Millionen Euro ein System geschaffen haben, mit dem etwa sie zwei Drittel aller VPNs und auch den Verkehr mit etwa einem Fünftel der populärsten `https`-Internetseiten abhören kann. Diese Spekulation ist konsistent mit Etatposten für die NSA.

Da diese Überlegung nun öffentlich ist, ist allerdings vielleicht genau der Fall eingetreten, vor dem in der zu Beginn erwähnten Präsentation des GCHQ gewarnt wird, und die Fähigkeiten sind schon wieder deutlich eingeschränkt worden.

4.4.3. Was man nicht vergessen sollte. Seit den Enthüllungen von Edward Snowden war die NSA in den Medien sehr präsent. In der Krypto-Szene gibt es eine lange Tradition, den obersten Knacker mit „NSA“ zu bezeichnen. Auch in diesem Abschnitt ging es wieder um die Fähigkeiten der NSA.

Hierbei sollte aber nicht vergessen werden, dass es neben den Vereinigten Staaten und ihren geheimdienstlichen Partnern Kanada, Großbritannien, Australien

[3] In dem Artikel wird das Faktorisierungsproblem behandelt. Das diskrete Logarithmusproblem kann mit einer ähnlichen Maschine angegriffen werden.

und Neuseeland noch weitere Länder mit erheblichen Mitteln gibt. Die USA, Kanada, Großbritannien, Australien und Neuseeland sind immerhin Rechtsstaaten, in welchen die Dienste klaren Beschränkungen und Kontrollen unterliegen, und lebhafte Demokratien.

4.5. Spezialistentum. Auf allen Gebieten der Wissenschaft führt der wissenschaftliche Fortschritt dazu, dass Forscher immer mehr Wissen haben müssen, um neue Arbeiten zu verstehen oder sogar, um nur ein grundlegendes Verständnis zu erlangen. Es gibt somit einen allgemeinen Trend zur Spezialisierung. Die Ausprägung der Informatik und dann der Kryptologie als Teildisziplin der Informatik sind Aspekte dieses Trends. Nun schreitet der Prozess innerhalb der Kryptologie voran.

Es gibt drei Aspekte des allgemeinen Trends zur Spezialisierung: Erstens bauen Arbeiten auf vorherigen Arbeiten auf; selbst wenn Ergebnisse nicht direkt benutzt werden, ist eine gewisse Vertrautheit mit Definitionen und Techniken notwendig, um eine Arbeit zu lesen. Dieses notwendige Hintergrundwissen wächst beständig an. Zweitens werden Arbeiten auch dann schwieriger zu lesen, wenn man mit dem Gebiet vertraut ist. Drittens werden nicht nur beständig neue Arbeiten der Literatur hinzugefügt, sondern die Anzahl der in einem Jahr veröffentlichten Arbeiten nimmt auch zu.

Heutzutage ist es selbst für Experten schwer, neue Arbeiten zu beurteilen. Um eine Arbeit ohne Vorbereitung zu lesen, muss sie im Allgemeinen sehr nahe an der persönlichen Forschung liegen und selbst dann muss man mit einigen Tagen Arbeit rechnen. Manchmal können auch Monate oder sogar ein Jahr an Vorbereitungen notwendig sein, bevor man eine Arbeit lesen und im Detail verstehen kann.

Diese Beschreibung passt auf viele Gebiete der Wissenschaft, aber für Kryptographie sind aufgrund der ureigenen Ziele der Kryptographie selbst die Implikationen grundlegend verschieden von denen für die reine Mathematik, um ein Beispiel zu nennen.

Warum erregte die offensichtliche Hintertür im Dual_EC_DRBG und in Implementationen wie denen von RSA Securities nicht schon vor den Enthüllungen von Edward Snowden Aufsehen? Zurückschauend stellt man fest, dass schon im Jahr 2007 auf der prominentesten kryptologischen Fachtagung, der CRYPTO, zwei Angestellte von Microsoft in einer informellen Präsentation auf die offensichtliche Hintertür hinwiesen ([32]). Trotz dieser Information scheint niemandem in den Sinn gekommen zu sein, zu überprüfen, ob die „Empfehlung“ von NIST implementiert worden war. Der Autor hat keine Erklärung hierfür außer vielleicht der, dass er noch nicht mal von der „Empfehlung“ und dem Generator gehört hatte, obwohl er auf genau dem Gebiet der elliptischen Kurven Kryptographie arbeitet, und auf viele seiner Kollegen trifft dies auch zu.

Das praxisorientierte Ziel des rigorosen, mathematischen Zugangs zu kryptographischen Verfahren basierend auf einem soliden Fundament, so wie in Abschnitt 3.2.5 beschrieben, ist es, Gewissheit in die Sicherheit des Verfahrens bei unerwarteten Angriffen zu haben. Wie am Ende von Abschnitt 3.2.5 erörtert, gibt es immer eine Lücke zwischen Theorie und Praxis, welche nicht leicht zu schließen ist. Leider geht die tatsächliche Ausführung des rigorosen Ansatzes zu kryptographischen

Verfahren mit weiteren Problemen einher:

Die genauen Aussagen in wissenschaftlichen Arbeiten der Kryptographie sind oftmals selbst für Wissenschaftler, die in dem Gebiet der Arbeiten forschen, schwer zu verstehen und zu interpretieren. Noch schwerer ist es, zu überprüfen, ob die angeblichen Beweise korrekt sind. Nicht nur dies, es kommt nicht zu selten vor, dass Arbeiten mit angeblichen reduktiven Sicherheitsresultaten nicht halten, was sie versprechen. Manchmal werden die genauen Beiträge in der Einleitung irreführend dargestellt; manchmal wird, um ein Sicherheitsresultat für ein bestimmtes Angriffsszenario zu erhalten, ein Verfahren entworfen, dass dann schwach bezüglich eines offensichtlichen Angriffs außerhalb des Szenarios ist; manchmal scheint das unterliegende rechnerische Problem ein gekünsteltes zu sein, dass nur erfunden wurde, um irgendeine Art von Resultat zu erlangen; und schließlich sind die angeblichen Beweise manchmal einfach falsch. Oftmals ist so ein Problem erst entdeckt worden, nachdem das Verfahren schon erfolgreich angegriffen worden war.

Die Tatsache, dass eine Anzahl von Protokollen, welche zuvor als „beweisbar sicher“ angepriesen worden waren (was tatsächlich bedeutet, dass ein reduktives Sicherheitsresultat angeblich erzielt worden war), unerwartete praxisrelevante Sicherheitslücken hatten, wurde von Neal Koblitz, dem Miterfinder der elliptischen Kurven Kryptographie, und Alfred Menezes kritisch beleuchtet. Koblitz und Menezes veröffentlichten ihre Kritik in einem ungewöhnlichen Artikel mit Titel „Another look at provable security“ im Journal of Cryptology ([21]) und weiteren Artikeln mit ähnlichen Titeln. „Genau wie mit anderen aufgebauschten Ideen – Atomschutzkellern in den 1950ern, Raketenabwehrschildern in den 1980ern – erzeugen ‚Beweise‘ für die Sicherheit eines kryptographischen Protokolls oftmals falsches Gefühl der Sicherheit, welches Leute blind macht für die wirklichen Gefahren“, so Koblitz ([20]).

Die Kritik von Koblitz und Menezes führte zu einem heftigen Disput, in welchem die konträre Position besonders von Oded Goldreich, dem Autor der „Foundations of Cryptography“ vertreten wurde ([15]). Seiner Meinung nach gibt es „in der Kryptographie leider allzu häufig irrige Vorstellungen über die Bedeutung von Resultaten. Koblitz und Menezes haben aber, außer, dass sie auf einige bekannte Fehler in veröffentlichten angeblichen Beweisen hingewiesen haben, nur zur Verwirrung beigetragen mit einem Artikel, der auch voller solcher irrigen Vorstellungen ist.“ Fehler, irrige Vorstellungen und Missverständnisse würden aber so oder so nur die Notwendigkeit eines wissenschaftlichen Zugangs zur Kryptographie unterstreichen, eines Zugangs, der auf rigorosen Begriffen und Analyse beruht ([16]).

Ohne weiter auf die Einschätzungen von Koblitz und Menezes sowie von Goldreich einzugehen möchte der Autor betonen, dass es der WISSENSCHAFTLICHE FORTSCHRITT SELBST ist, der mit zunehmender Spezialisierung einhergeht, welche wiederum damit einhergeht, dass viele wichtige Arbeiten nur von einer sehr kleinen Anzahl von Leuten gelesen und durchdacht werden können. Da Menschen nur begrenzte Zeit zur Verfügung haben und in der Tat irren, gibt es kein einfaches Heilmittel gegen die Gefahr von irreführenden und falschen Aussagen und Publikationen oder irrigen Vorstellungen von Lesern.

4.6. Das große Bild. Nach jahrhundertelangen Dasein im Halbschatten und im Einfluss von Mächtigen ist die Kryptologie nun in vielerlei Hinsicht in die Öffentlichkeit getreten.

Es gibt eine aktive Forschung mit weltweit öffentlichen Ergebnissen, es gibt etablierte wissenschaftliche Prinzipien, einen nicht abreißenden Strom neuer Resultate, Ideen für neue Anwendungen sowie einen technischen Fortschritt, der die neuen Anwendungen möglich macht. Kryptographische Verfahren wie Krypto-Währungen oder Krypto-Verträge könnten große Auswirkungen auf Bereiche der Wirtschaft oder sogar die Gesellschaft als Ganzes haben.

Wie über die vergangenen Jahrhunderte stellt sich aber nach wie vor die Frage, ob die verwendeten Verfahren und Produkte in der täglichen Anwendung wirklich sicher sind.

Der Laie muss sich hier wie in anderen Bereichen des modernen Lebens auf Spezialisten verlassen, Spezialisten, die auch ein begrenztes Wissen haben, Fehler machen können oder auch andere Interessen als die vorgegebenen haben können. Wie kann hier zumindest partiell Abhilfe geschaffen werden?

Nun, ein einzelner Spezialist mag irreführende Aussagen machen oder auch von Laien falsch verstanden werden. Falsche Aussagen von einzelnen sind aber relativ irrelevant, wenn es einen Prozess gibt, in welchem schlechte Ideen und Verfahren verworfen werden und sich bessere Ideen durchsetzen.

Wie auch in anderen Bereichen gilt auch in der Kryptologie: Es sind die richtigen gesellschaftlichen Institutionen, auf denen der Fortschritt beruht. Werte wie Integrität, Selbstkritik und Offenheit gegenüber Neuem, ein von Transparenz, Sachlichkeit, Kooperation und Wettbewerb geprägter Umgang sowie die richtigen formalen Organisationen mit klaren abgegrenzten Zielen frei von Interessenskonflikten führen in einem kontinuierlichen Verbesserungsprozess zu guten Ideen und Produkten.

Literatur

[1] David Adrian, Karthikeyan Bhargavan, Zakir Durumeric, Pierrick Gaudry, Matthew Green, J. Alex Halderman, Nadia Heninger, Drew Springall, Emmanuel Thomé, Luke Valenta, Benjamin VanderSloot, Eric Wustrow, Santiago Zanella-Béguelin und Paul Zimmermann, Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice, In *Procedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.

[2] Daniel Bernstein und Tanja Lange, Faster Addition and Doubling on Elliptic curves. In *Advances in Cryptology – ASIACRYPT 2007*, Springer, Berlin, 2007, 29–50.

[3] Daniel Bernstein, Tanja Lange und Ruben Niederhagen, Dual EC: A Standardized Back Door. Cryptology ePrint Archive: Report 2015/767, 2015.

[4] Bundesamt für Sicherheit in der Informationstechnik, BSI Technische Richtlinie TR-02102-1, Kryptographische Verfahren: Empfehlungen und Schlüssellängen, 2015.

[5] Anne Canteaut, Céderic Lauradoux und André Seznec, Understanding cache attacks. INRIA Rapport de recherche No. 5881, 2006.

[6] Richard Clarke, Michael Morell, Geoffrey Stone, Cass Sunstein und Peter Swire, Liberty and Security in a Changing World, Report and Recommendations of The President's Review Group on Intelligence and Communications Technologies. The White House, 2013.

[7] Whitfield Diffie und Martin Hellman, New Directions in Cryptography. *IEEE Transactions on Information Theory* **2** (1976), 644 – 654.

[8] Harold Edwards, A normal form for elliptic curves. *Bulletin of the American Mathematical Society* **44** (2007), 393–422.

[9] Jens Franke, Thorsten Kleinjung, Christof Paar, Jan Pelzl, Christine Priplata und Colin Stahlke, SHARK. Presented at SHARCS - Special-purpose Hardware for Attacking Cryptographic Systems 2005. `http://www.sharcs.org`

[10] Steven Galbraith, Mathematics of Public Key Cryptography. Cambridge University Press, Cambridge, UK, 2012.

[11] Carl Friedrich Gauß, Disquisitiones Arithmeticae. Gerhard Fleischer, Leipzig, 1801.

[12] GCHQ, BULLRUN. Internal presentation. `http://www.spiegel.de/media/media-35532.pdf`

[13] Kristian Gjøsteen, Comments on the Dual-EC-DRBG / NIST SP 800-900, 2005. `http://www.math.ntnu.no/~kristiag/drafts/dual-ec-drbg-comments.pdf`

[14] Oded Goldreich, Foundations of Cryptography I und II. Cambridge University Press, Cambridge, UK, 2001 und 2004.

[15] Oded Goldreich, On Post-Modern Cryptography. Cryptology ePrint Archive: Report 2006/461, 2006.

[16] Oded Goldreich, Conversation with the author, 2016.

[17] Godfrey Harold Hardy, A Mathematician's Apology. Cambridge University Press, Cambridge, UK, 1940.

[18] David Kahn, The Codebreakers – The Comprehensive History of Secret Communication from Ancient Times to the Internet. Scribner, New York, 1996.

[19] Auguste Kerckhoffs, La cryptographie militaire. *Journal des sciences militaires* **9** (1883), 5–38 & 161–191.

[20] Neal Koblitz, The uneasy relationship between mathematics and cryptography. *Notices of the AMS* **54** (2007), 972 – 979.

[21] Neal Koblitz und Alfred Menezes, Another Look at Provable Security. *Journal of Cryptology* **20** (2007), 3 – 37.

[22] Maurice Kraitchik, Théorie des Nombres, Gauthier-Villars, Paris, 1922.

[23] Joseph Menn, Exclusive: Secret contract tied NSA and security industry pioneer. Reuters, Dec. 20 2013.

[24] Mohammed Mrayati, Y. Meer Alam und M. H. Tayyan, Series on Arabic Origins of Cryptology 1 – 3. KFCRIS & KACST, Riyadh, 2002 – 2003

[25] Ellen Nakashima, National Security Agency plans major reorganization. Washington Post, Feb. 2 2016.

[26] National Security Agency, Suite B Cryptography, Aug. 19 2015. `https://www.nsa.gov/ia/programs/suiteb_cryptography/index.shtml`

[27] New York Times, Secret Documents Reveal N.S.A. Campaign Against Encryption, Sep. 5 2013.

[28] Ron Rivest, Adi Shamir, Leonard Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM* **21** (1987), 120–126.

[29] RSA Security, RSA response to media claims regarding NSA relationship, 2013. `https://blogs.rsa.com/rsa-response`

[30] Klaus Schmeh, Codeknacker und Codemacher, W3L, Bochum, 2014.

[31] Shannon, Claude, Communication Theory of Secrecy Systems. Bell System Technical Journal **28** (1948), 656 – 715.

[32] Dan Shumow und Niels Ferguson, On the Possibility of a Back Door in the NIST SP800-90 Dual EC Prng. CRYPTO Rump Session, 2007. `http://rump2007.cr.yp.to/15-shumow.pdf`

[33] Peter Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Journal on Computing* **26** (1997), 1484–1509.

[34] Maurice Wilkes, Time-Sharing Computer Systems. American Elsevier, 1968.

Claus Diem, Mathematisches Institut, Universität Leipzig, Leipzig, Deutschland
E-mail: diem@math.uni-leipzig.de