1. Einleitung

Die Welt wie wir sie kennen, befindet sich, was den Informationsaustausch angeht, vor einem Umbruch. Das Internet, das Netz der Netze, ermöglicht uns den Zugriff auf Terrabytes an Informationen und, vielleicht sogar noch wichtiger, die weltweite Kommunikation mit anderen Menschen, zum Ortstarif. Schon heute werden weltweit mehr Emails verschickt als Telefongespräche geführt.
Doch was die wenigsten wissen: Diese Kommunikation kann noch leichter abgehört werden als ein Telefongespräch. Das amerikanische Abhörsystem ECHELON analysiert täglich weit über zwei Millionen Emails. Das stellt eine massive Bedrohung der Privatsphäre dar. Doch gibt uns die Mathematik Methoden in die Hand, uns solcher Angriffe zu erwehren: Die Kryptologie.
Die Wissenschaft vom Verbergen, so die Übersetzung aus dem Griechischen, ist so alt wie die Kultur des Abendlandes. Schon der Spartanerkönig Leonidas lies die Botschaften seiner Spione durch Umstellen der Buchstaben verschlüsseln (man nennt dies eine Transpositionschiffre). Mit dem Beginn der modernen Großreiche im 17. Jahrhundert erlebte die Kryptologie ein neues Hoch, und auch in der heutigen Zeit ist das Thema „Verschlüsselung von Nachrichten“ wieder aktueller als jemals zuvor.
Im folgenden sollen die mathematischen Hintergründe einiger kryptologischer Verfahren ebenso betrachtet werden wie der praktische Einsatz dieser Verfahren.

2. Begriffsbestimmung

Die Kryptologie hat drei verschieden Ziele. Zum ersten natürlich die Geheimhaltung: Außer dem berechtigten Empfänger sollte es allen unmöglich sein, die Nachricht lesen zu können. Genau dieser Teilbereich ist der bekannteste und derjenige, der am meisten mit dem Begriff „Kryptologie“ in Verbindung gebracht wird. Doch ist dies vielleicht nicht einmal der wichtigste Teilbereich. Mindestens genauso wichtig ist es, die Integrität einer Nachricht zu sichern. Die geschieht mittels raffinierter Prüfsummenverfahren und stellt mit das interessanteste Teilgebiet der Kryptologie dar. Als letztes versucht man, mittels kryptologischer Verfahren auch die Identität eines Menschen zu verifizieren, bei einem kryptographischen System spricht man dabei von der Benutzerauthentikation.

Man teilt die Kryptologie in zwei Teilgebiete: Die Kryptographie und die Kryptanalyse. Die oben vorgestellten Ziele erreicht man mittels kryptographischen Methoden, denn die Kryptographie beschäftigt sich, generell gesagt, lediglich mit dem Verschlüsseln von Nachrichten. Das zweite, fast schon wichtigere - allerdings mathematisch gesehen auch wesentlich anspruchsvollere - Gebiet ist das der Kryptanalyse. Die Kryptanalyse prüft die Sicherheit kryptographischer Systeme, sie versucht die Sicherheit derselben zu beweisen – oder eben auch deren Unsicherheit, indem sie die verwendeten Algorithmen und Protokolle bricht.

Ein kryptographisches System besteht aus einer Reihe von hintereinander anzuwendenden Protokollen. Diese Regeln z.B. den Schlüsselaustausch unter den einzelnen Teilnehmern, legen fest, wie Nachrichten zu senden sind etc. Dabei bedient sich jedes Protokoll – quasi als „Handwerkszeug“ – einer Reihe von Algorithmen, um sein Ziel zu erreichen. So braucht etwa ein Protokoll zur Benutzerauthentikation einen asymmetrischen Verschlüsselungsalgorithmus. Es gibt zwei Arten von Verschlüsselungsalgorithmen: Die symmetrischen und die asymmetrischen. Symmetrische Verschlüsselungen ver- und entschlüsseln (sie chiffrieren und dechiffrieren) mit einem Schlüssel, der dann natürlich sowohl dem Empfänger als auch dem Sender bekannt sein muß. Dabei entsteht das Risiko, das ein unberechtigter dritter in den Besitz des Schlüssels gelangen und somit die gesamte Kommunikation abhören könnte. Ein eindeutiger Vorteil dieser Algorithmen ist jedoch ihre Geschwindigkeit: Aufgrund geringerer mathematischer Komplexität (sie basieren zumeist auf linearen Schieberegistern, Permutationen und einer Anzahl binärer, in Hardware sehr leicht zu realisierender Rechenoperationen) erreicht man Verschlüsselungsraten, die um den Faktor 100 größer sein können als die mit asymmetrischen Algorithmen zu erreichende Geschwindigkeit.

Asymmetrische Algorithmen verwenden statt einem vier Schlüssel. Jeder Teilnehmer an einem solchen System hat einen öffentlichen und einen geheimen Schlüssel. Nachrichten an ihn werden mit dem öffentlichen Schlüssel chiffriert, können aber nur mittels des privaten Schlüssels wieder dechiffriert werden. Dabei muß natürlich sichergestellt werden, daß man den einen der beiden Schlüssel nicht aus dem anderen berechnen kann. Diese Algorithmen sind aufgrund ihrer Komplexität allerdings langsam. In der Praxis setzen sich Hybridsysteme durch: Es wird ein zufälliger Sitzungsschlüssel erzeugt und die Nachricht mit diesem unter Verwendung eins symmetrischen Algorithmus verschlüsselt. Der Sitzungsschlüssel selbst wird mit dem öffentlichen Schlüssel des Empfängers chiffriert – natürlich diesmal mittels eines asymmetrischen Verfahrens – , und an die codierte Nachricht angehängt. Ein gutes Beispiel hierfür ist die beliebe Verschlüsselungssoftware PGP: Sie verschlüsselt die Nachricht

mit dem relativ neuen Algorithmus IDEA und den Schlüssel mit dem bekanntesten asymmetrischen Verfahren überhaupt, dem RSA – Verfahren.
Die Kryptologie soll sicherstellen, daß nur ein berechtigter Empfänger in der Lage ist, eine Botschaft zu dechiffrieren. Die Dechiffrierung ohne den Schlüssel soll ein möglichst schweres Problem sein, manchmal, wenn Informationen auf unbestimmte Zeit hin geheimgehalten werden sollen, sollte es sogar ein unlösbares Problem sein. Die Mathematik kennt seit langem die Einteilung mathematischer Probleme in verschieden Klassen von Problemen. Dabei geht es darum, den jeweils schwierigsten Einzelfall einer Klasse von Problemen zu lösen. Die Lösung kann dabei mittels verschiedener Algorithmen erfolgen. Kann ein Algorithmus eingesetzt werden, bei dem die Anzahl der Rechenschritte sich polynomial zur Anzahl der zu berechnenden Einzelwerte verhält, heißt das Problem berechenbar. Natürlich kann auch der Aufwand zur Berechnung eines solchen Problems zu groß sein, um in der Praxis angewendet zu werden. allerdings geht man bei dieser Einteilung auch nicht von Computern im heutigen Sinne aus, sondern vielmehr von einer sogenannten Turing - Maschine. Dabei handelt es sich um einen theoretischen Computer mit unendlich großem Speicher und nicht näher bestimmter Arbeitsgeschwindigkeit (auf jeden Fall einer sehr, sehr hohen). Dieser Computer kann berechenbare Probleme aufgrund seiner hohen Verarbeitungskapazität auf jeden Fall in akzeptabler Zeit lösen. In polynomialer Zeit berechenbare Probleme heißen manchmal auch schwer, meistens aber einfach die Klasse P der Probleme. Hier ist ein Ausflug in die Komplexitätstheorie notwendig: Dort teilt man Probleme nach ihrer Komplexität ein, und die Klasse P steht dabei ganz unten. Darüber steht die Klasse der NP – Probleme, die ebenfalls in polynomialer Zeit gelöst werden können – allerdings nur auf einer nichtdeterministischen Turing - Maschine, d.h. einer Turing – Maschine die die Lösung des Problems errät und alle Vermutungen in polynomialer Zeit überprüft. Mit heutigen Mitteln sind NP – Probleme nicht in polynomialer Zeit zu lösen. Die Bedeutung dieser Einteilung für die Kryptologie ist die folgende: Das Verschlüsseln eines Algorithmus sollte ein P – Problem darstellen, d.h. es sollte in polynomialer Zeit möglich sein die Nachricht zu verschlüsseln. Ebenso muß der berechtigte Empfänger in Kenntnis des Schlüssels in der Lage sein, die Nachricht in polynomialer Zeit zu entschlüsseln. Den unberechtigten Empfänger soll die Chiffre, d.h. der angewandte Algorithmus vor ein Problem der Klasse NP stellen – er darf nicht in der Lage sein, den Code mittels eines realisierbaren Computers zu entschlüsseln (die Turing – Maschine kann *per definitionem* nicht verwirklicht werden, hier befindet sich die Kryptologie also auf der sicheren Seite). Man kann beweisen, daß beliebige Probleme aus der Klasse NP so schwierig sind wie jedes andere Problem der Klasse NP. Solche Probleme heißen NP – vollständige

Probleme. Normalerweise nimmt man zur Verschlüsselung von Nachrichten heute NP – vollständige Probleme. Dies birgt allerdings ein gewisses Risiko: Sollte jemals bewiesen werden, daß P = NP ist (bis heute ist dies die zentrale Kernfrage der Komplexitätstheorie, niemand erwartet ernstlich eine Beantwortung dieser Frage in der nächsten Zeit), so könnten diese Verschlüsselungen mit deterministischen Algorithmen in polynomialer Zeit gebrochen werden. Die beiden anderen Klassen von Problemen, PSPACE (lösbar in polynomialer Zeit auf einer nichtdeterministischen Turing – Maschine mit polynomialem Speicherplatz) und EXPTIME (nur in exponentieller Zeit lösbar) spielen für die Kryptologie keine besondere Rolle.

3. Historische Beispiel für Algorithmen und deren Kryptanalyse

3.1. Einfache monoalphabetische Chiffren[1]

Es gibt im Grunde genommen lediglich zwei Arten monoalphabetischer Chiffren. Zum einen sind dies die sogenannten Transpositionschiffren. Diese werden auch als Permutationen bezeichnet. Bei einer Transpositionschiffre werden die Positionen der einzelnen Buchstaben verändert. Ein gutes Beispiel dafür ist die Skytale, eine von den Griechen und Spartanern verwendete Holzrolle. Die Botschaft wurde auf Pergament geschrieben, dieses in Streifen geschnitten und einzeln um die Skytale gewickelt. Der auf der Skytale lesbare Text war nicht zu verstehen, er enthielt zum Beispiel jeden sechsten Buchstaben der Nachricht, immer 6 Buchstaben hintereinander, danach begann die nächste Reihe. Transpositionschiffren, allerdings wesentlich ausgefeilter als diese, werden auch heute noch oft verwendet. Viele symmetrische Algorithmen wie z.B. DES oder IDEA verwenden Transpositionschiffren als Eingangs- und Ausgangspermutationen. Die berühmte Verschlüsselungsmaschine der Deutschen im zweiten Weltkrieg, die Enigma, realisierte eine ganz besonders komplizierte Transpositionschiffre.
Die zweite Art von monoalphabetischen Chiffren sind die auf Substitution einzelner Buchstaben beruhenden. Diese Technik ist ebenfalls sehr alt: Schon Cäsar verwendete diese Technik, um geheime Briefe abzufassen. Er verwendete eine regelmäßige Chiffre: Jeder Buchstabe wurde durch den Buchstaben des Alphabets ersetzt, der drei Stellen weiter hinten

[1] nach [1], Seite 18-23

stand. So wurde aus einem A ein D, aus einem B ein E und aus einem I ein L. Allgemein handelt es sich um eine Chiffre, bei der Buchstaben um einen bestimmten Wert n im Alphabet verschoben werden. Bei der Cäsar - Chiffre handelt es sich um n = 3, aber es gibt natürlich 26 verschiedene Möglichkeiten, auch die triviale Verschlüsselung A -> A, B -> B etc. Natürlich kann die Wahl der Substitutionen für die einzelnen Buchstaben auch völlig willkürlich sein. Dies erschwert dann die Kryptanalyse. Man kann z.B. ein Schlüsselwort ausmachen. Nehmen wir an, das Schlüsselwort sei ANGEL. Dann schreibe ich zwei Alphabete, ein Klartext – Alphabet und ein Geheim – Alphabet nach dem folgenden Rezept:

Klartextalphabet: a b c d e f g h i j k l m n o p q r s t u v w x y z
Geheimalphabet: a n g e l h i j k m o p q r s t u v w x y z b c d f

Mit dieser Methode kann man es vermeiden, ständig neue Alphabete als Ganzes verschicken zu müssen, der Austausch eines Schlüssels genügt dann.

3.2. Kryptanalyse einer monoalphabetischen Substitutionschiffre[2]

Für einen Kryptanalytiker haben lebende, gesprochene Sprachen zwei ungeheure Vorteile: Zum einen ergeben die meisten Kombinationen von Buchstaben lediglich ein sinnloses Wirrwarr, zum anderen tauchen in längeren Texten Buchstaben stets mit einer bestimmten Wahrscheinlichkeit auf. So ist zum Beispiel in der deutschen Sprache der häufigste Buchstabe das e. Die genaue Häufigkeitsverteilung ist der folgenden Tabelle zu entnehmen:

Buchstabe	***Häufigkeit***	***Buchstabe***	***Häufigkeit***
a	6,51	n	9,78
b	1,89	o	2,51
c	3,06	p	0,79
d	8,08	q	0,02
e	17,4	r	7
f	1,66	s	7,27
g	30,1	t	6,15
h	4,76	u	4,35
i	7,55	v	0,67

[2] nach [1], Seite 23-26

j	0,27	w	1,89
k	1,21	x	0,03
l	6,44	y	0,04
m	2,53	z	1,13

Wenn man nun also im Geheimtext die Buchstaben zählt, so kann man, wenn der Text nicht zu kurz ist, ohne weiteres die Buchstaben e,n,d,i,s,t und a identifizieren. Das hilft einem bereits wesentlich weiter. Um die Kryptanalyse zu vervollständigen, ist das jedoch noch nicht genug. Man muß nämlich auch die Buchstabenpaare berücksichtigen. Diese haben folgende Häufigkeitsverteilung:

Buchstabenfolge	***Häufigkeit in %***
en	3,88
er	3,75
ch	2,75
te	2,26
de	2
nd	1,99
ei	1,88
ie	1,79
in	1,67
es	1,52

Somit ist die Kryptanalyse schon fast vollendet. Wenn ich jetzt zum Beispiel das Paar „ch“ noch daran identifiziere, daß die Geheimäquivalente von c und h sehr oft als „ch“ auftauchen, aber so gut wie niemals als „hc“, so kann ich die Kryptanalyse vollenden. Fehlen noch ein paar Buchstaben, so kommt uns die Redundanz der lebenden Sprachen zu Hilfe: Wörter und Zusammenhänge können auch dann noch erkannt werden, wenn wirklich ein großer Teil der Buchstaben nicht bekannt ist. Diese „statistischen Attacken“ auf Chiffretexte, man spricht von einer „known – ciphertext – attack“, sind so gut und genau, daß man die heute monoalphabetische Chiffren leicht von Computern in automatisierten Verfahren brechen kann, die kaum noch der Mitwirkung eines Menschen bedürfen.

4. Polyalphabetische Chiffren[3]

Wie oben dargelegt, ist das Problem einer monoalphabetischen Chiffre, daß die Häufigkeiten der einzelnen Buchstaben es leicht machen, sie zu brechen. Ein logischer Schritt ist es nun, einzelnen Buchstaben mehrere Äquivalente zuzuweisen. Der Buchstabe e, der besonders häufig vorkommt, muß dann die meisten Äquivalente erhalten, der Buchstabe J wohl nur eines. Insgesamt sollten die Geheimäquivalente der Buchstaben alle mit der gleichen Häufigkeit im Geheimtext auftauchen. Man kann zum Beispiel den 26 Buchstaben des Alphabets die Zahlen von 00 bis 99 zuweisen. Das macht eine Kryptanalyse nach oben beschriebenem Verfahren unmöglich. Trotzdem hat ein Kryptanalytiker die Möglichkeit, auch solche Texte zu entschlüsseln: Er orientiert sich dabei einfach an der Häufigkeit der einzelnen Buchstabenpaare und an der Verteilung der einzelnen Buchstaben auf die Worte. Allerdings ist diese Art der Kryptanalyse anspruchsvoller, in den meisten Fällen muß der Mensch dem Computer ein bißchen helfen.

4.1. Die Vernam - oder Vigenère – Chiffre

Eine ganz besonders berühmte polyalphabetische Chiffre ist die sogenannte Vernam – oder Vigenère – Chiffre. Die im obigen Absatz vorgestellte Verschlüsselung macht den Austausch der Schlüssel sehr schwer: Aufgrund ihrer Eigenschaft, die Häufigkeiten der einzelnen Buchstaben zu nivellieren, muß der Schlüssel mindestens so lange sein wie die Summe der Geheimtext – Substitutionen plus der 26 Buchstaben des Alphabets. Im obigen Beispiel wäre der Schlüssel mit 126 Symbolen, oder aber, wenn wir 00 bis 99 als je zwei Symbole rechnen, 226 Symbolen, doch sehr lang. Da man immer davon ausgehen muß, daß Schlüssel kompromittiert werden, versucht man, sie möglichst kurz zu halten, um sie leichter und vor allem unauffälliger übermitteln zu können.
Der französische Diplomat Blaise de Vigenère erfand 1586 ein Chiffresystem, dessen Grundidee so einfach und logisch wie zugleich effektiv ist: Die Aneinanderreihung mehrerer monoalphabetischer Chiffrierungen. Um eine Vigenère – Chiffre zu erzeugen, braucht man zum einen das Schlüsselwort und zum anderen ein Vigenère – Quadrat. Dieses sieht folgendermaßen aus:

[3] nach [1], Seite 35 – 38, ebenso 4.1.

a	***b***	***c***	***d***	***e***	***f***	***g***	***h***	***i***	***j***	***k***	***l***	***m***	***n***	***o***	***p***	***q***	***r***	***s***	***r***	***u***	***v***	***w***	***x***	***y***	***z***
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	X	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	w
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Die Verwendung dieses Quadrates ist denkbar einfach: Man schreibt die Botschaft und darunter reiht man das Schlüsselwort immer wieder aneinander (Leerzeichen werden dabei ignoriert). Dann sucht man sich den Buchstaben der Botschaft, der gerade verschlüsselt werden soll, und sucht ihn in der ersten Zeile. Dann geht man in der Spalte soweit hinunter,

bis man ganz links den aktuellen Buchstaben des Schlüsselwortes gefunden hat. Zum Beispiel ergibt sich aus dem Klartextbuchstaben „P“ und dem Schlüsselwortbuchstaben „V“ der Geheimtextbuchstabe „K“. Es ist klar, daß die Häufigkeit der einzelnen Buchstaben viel gleichmäßiger verteilt ist als bei einer monoalphabetischen Chiffre, denn der Klartextbuchstabe „E“ kann auf alle anderen 25 Buchstaben des Alphabets abgebildet werden. Gleichzeitig ist es auch sehr einfach, mit dieser Methode anderen Botschaften zukommen zu lassen, denn das Schlüsselwort muß nicht allzu lang sein, 5 bis 10 Buchstaben genügen – die oben vorgestellte Methode brauchte z.B. 226 Zeichen.

4.2. Kryptanalyse der Vigenère – Chiffre – Kassiski - und Friedman - Test[4]

Der Angriff auf eine mit der Vigenère – Chiffre verschlüsselte Botschaft nutzt die in einem längeren Text vorhandenen statistischen Regelmäßigkeiten und Merkmale nicht dazu aus, den Inhalt der Nachricht direkt zu entschlüsseln. Ziel ist es vielmehr, das Schlüsselwort offenzulegen. Dabei helfen einem zwei Methoden, die, obwohl beide nicht mehr taufrisch, dennoch extrem wichtig sind, und das nicht nur zur Entschlüsselung der Vigenère - Chiffre, sondern zum Umgang mit polyalphabetischen Chiffren allgemein.
Der sogenannte Kassiski – Test wurde nach dem preußischen Infanteriemajor Friedrich Wilhelm Kassiski (1805 – 1881) benannt. Entwickelt wurde er allerdings schon 1854 von Charles Babbage, einem englischen Mathematiker, der den Grundgedanken zur Konzeption eines modernen Computers hatte.
Betrachtet man das Schema der Vigenère – Verschlüsselung. so fällt einem auf, daß, bei gegebenem Schlüsselwort der Länge n, alle k* n Buchstaben, wobei k eine natürliche Zahl sein soll, mit dem selben Buchstaben des Schlüsselwortes verschlüsselt werden. Bei der Kryptanalyse nun dreht man die Sache um: Wenn ich im Geheimtext Folgen gleicher Buchstaben finde (und zwar nicht nur zwei, sonder schon drei bis fünf) finde, so kann ich davon ausgehen, daß ihr Abstand ein Vielfaches der Länge n des Schlüsselwortes entspricht. Also empfiehlt es sich, den gesamten Text auf diese Weise zu analysieren und die gefundenen Abstände in Primfaktoren zu zerlegen. Die kleinste Kombination der Primfaktoren ist dann mit großer Wahrscheinlichkeit die Länge des Schlüsselwortes. Der Kassiski – Test funktioniert in der Praxis so gut, daß man kaum jemals mit dem Ergebnis daneben liegt – so berechenbar sind natürliche Sprachen. Allerdings muß man berücksichtigen, daß der Kassiski – Test die Schlüsselwortlänge natürlich nur bis auf Vielfache oder Teiler bestimmt. Deswegen

gibt es einen weiteren Test, der dazu dient, die Größenordnung der Schlüsselwortlänge zu bestimmen.

Dieser Test wurde nach Colonel William Frederick Friedman (1891 – 1969) benannt.

Greift man aus einem in einer natürlichen Sprache geschriebenen Text zufällig zwei Buchstaben heraus, so kann man mit einer bestimmten Wahrscheinlichkeit annehmen, die selben erwischt zu habe, das soll heißen, zweimal den Buchstaben „a“ etc. Diese Wahrscheinlichkeit läßt sich berechnen und ist als sogenannter Koinzidenzindex bekannt. Der Koinzidenzindex kann dabei helfen, zum einen mono- von polyalphabetischen Chiffren zu unterscheiden, zum anderen liefert er die grobe Größenordnung der Schlüsselwortlänge.

Die Vorgehensweise ist dabei denkbar einfach: Man stelle sich einen Text der Länge n vor. Dabei geben n_1 die Anzahl der Buchstaben „a“, n_2 die Anzahl der Buchstaben „b“ etc. an. Man interessiert sich nun für die Anzahl der Paare, bei denen beide Buchstaben „a“ gleich sind. Es gibt n_1 Möglichkeiten für die Auswahl des ersten Buchstabens, n_1-1 Möglichkeiten für die Auswahl des zweiten Buchstabens aus dem Text. Die Reihenfolge der Buchstaben spielt dabei keine Rolle. Es gibt also

$$z = \frac{n_1(n_1 - 1)}{2}$$

Möglichkeiten, Paare der beschriebenen Art auszuwählen. In Summenschreibweise für alle 26 Buchstaben ergibt sich logischerweise:

$$z_p = \sum_{i=1}^{26} \frac{n_i(n_i - 1)}{2}$$

Die Chance, ein Paar gleicher Buchstaben zu erwischen ist nach dem Zählprinzip:

$$P(X) = \frac{z_p}{z}$$

[4] nach [1], Seite 38ff.

P(X) wird als Friedmanscher Koinzidenzindex bezeichnet und mit dem Buchstaben I abgekürzt: I := P(X).
Stellt man sich nun umgekehrt vor, man würde die Wahrscheinlichkeit kennen, mit der ein bestimmter Buchstabe an einer bestimmten Stelle auftaucht. Dann wäre die Wahrscheinlichkeit, zweimal identische Buchstaben zu wählen, ungefähr $p_i * p_i$. Ungefähr deswegen, weil in dieser Rechnung der Buchstabe zweimal von der selben Stelle kommen dürfte, aber der Fehler geht bei großen n unter. Auf dies Art ausgerechnet ergibt sich die Näherung für den Koinzidenzindex eines Textes bei gegebenen Wahrscheinlichkeiten p_i der Buchstaben zu

$$I = \sum_{i=1}^{26} p_i^2$$

Für einen Text in deutscher Sprache ergibt sich I=0,0762. Wenn alle Buchstaben gleich häufig vorkommen (eine ideale polyalphabetische Chiffre könnte das bewirken), ergibt sich ein I von ungefähr 0,0385. Der Koinzidenzindex wird also größer, wenn die Buchstaben im Text unregelmäßiger verteilt sind. Dabei ist der Wert 0,0385 das absolute Minimum für den Koinzidenzindex.
Somit wäre ein Test gefunden, der uns hilft, mono- und polyalphabetische Chiffren zu unterscheiden. Allerdings kann man noch mehr: Wenn man sich das Schema der Vigenère-Chiffre genau ansieht, so erkennt man, daß bei einem Schlüsselwort der Länge l stets l Buchstaben mit einer normalen monoalphabetischen Chiffrierung verschlüsselt wurden. Für Jeweils l Buchstaben ist I also rund 0,0762. Wenn man allerdings Buchstaben nimmt, die die nicht beide im Bereich 1-*l*,l+1-2*l*,.... liegen, sinkt I wieder auf einen niedrigeren Wert, Stellen wir uns nun den verschlüsselten Text in Tabellenform vor: Die Anzahl der Spalten sei l, die Anzahl der Zeilen n/l (Rundungsfehler sind zu vernachlässigen). Wenn man aus den n Möglichkeiten, einen Buchstaben zu wählen, sich für eine entscheidet, so liegt gleichzeitig auch die Spalte und die Zeile fest, in welcher er sich befindet (man beachte, daß Buchstaben in der selben Zeile mit dem selben Schlüsselwortbuchstaben verschlüsselt wurden – sie können also wie eine monoalphabetische Chiffre behandelt werden!). Der gewählte Buchstabe liegt in einer Spalte, in der es noch n/l-1 andere Buchstaben gibt. Die Anzahl der Paare ist also

$$\frac{n(\frac{n}{l}-1)}{2} = \frac{n(n-l)}{2l}$$

Die Anzahl der Buchstaben, die nicht in der selben Spalte wie ein gegebener Buchstabe liegen, ergibt sich aus der Anzahl aller Buchstaben minus der Anzahl der Buchstaben aus der gleichen Spalte: n-n/l. Dann ist die Anzahl der Paare aus nicht gleichen Spalten bestimmt zu:

$$\frac{n(n-\frac{n}{l})}{2} = \frac{n^2(l-1)}{2l}$$

Die Anzahl der zu erwartenden Paare insgesamt, daß heißt Paare aus der selben Spalte und Paare aus verschiedenen Spalten ergibt sich, indem man die Anzahl der möglichen Paare mit ihren Wahrscheinlichkeiten multipliziert – es handelt sich im Grunde genommen ja nur um zwei einfache Bernoulli – Ketten. Dabei ist die Wahrscheinlichkeit für Paare aus der selben Spalte 0,0762 - in einem normalen, monoalphabetische verschlüsselten Text. Die Wahrscheinlichkeit für Paare aus verschiedenen Spalten ist die für eine polyalphabetische Verschlüsselung. Der Einfachheit halber gehe ich von einem völlig willkürlichen Schlüsselwort aus, so daß die Wahrscheinlichkeit für Paare hier exakt 0,0385 liegen soll. Also ist die Anzahl A der zu erwartenden Paare gleich:

$$A = \frac{n(n-l)}{2} \cdot 0{,}0762 + \frac{n^2(l-1)}{2l} * 0{,}0385$$

Nun wird wiederum das Zählprinzip angewendet: Anzahl der zu erwartenden Paare geteilt durch die Anzahl aller Paare:

$$\frac{A}{z_p} = \frac{n-l}{l(n-1)} \cdot 0{,}0762 + \frac{n(l-1)}{l(n-1)} \cdot 0{,}0385 = \frac{1}{l(n-1)} \cdot [0{,}0377n + l(0{,}0385n - 0{,}0762)]$$

Was genau hat man hier berechnet? Man hat die Wahrscheinlichkeit berechnet, mit der zwei willkürlich aus einem Text herausgegriffene Buchstaben gleich sind. Dies sollte einem bekannt vorkommen: Es stellt eine Näherung des Koinzidenzindexes dar. Also gilt dann auch:

$$I \approx \frac{0{,}0377n}{l(n-1)} + \frac{0{,}0385n - 0{,}0762}{n-1}$$

Es ist ein Leichtes, diese Formel nach l aufzulösen. Man erhält dann die auf Friedman zurückgehende Formel:

$$l \approx \frac{0{,}0377n}{(n-1)I - 0{,}0385n + 0{,}0762}$$

Das konkrete Vorgehen beim Dechiffrieren einer Vigenère – Chiffre ist dann: Mittels Kassiski – Test die Länge des Schlüsselworts bis auf Teiler bzw. Vielfache herausfinden. Die Größenordnung des Schlüsselwortes durch den Friedman – Test bestimmen. Sobald die Schlüsselwortlänge bekannt ist, die Buchstaben in gleichen Spalten (vorher natürlich in eine Tabelle wie oben beschrieben einordnen!) wie eine monoalphabetische Chiffrierung behandeln, man kann somit dann leicht das Schlüsselwort herausfinden (einfach in der Tabelle nachschauen!!).
Schwieriger wird es, wenn das Schlüsselwort so lange ist wie die Botschaft. Wenn dann noch ein absolut zufälliges Schlüsselwort gewählt wurde, handelt es sich um ein sogar theoretisch sicheres System: Ein one – time – pad, bei dem es so viele Möglichkeiten der Dechiffrierung wie verschiedene Botschaften gibt: Ein hoffnungsloses Unterfangen, so etwas dechiffrieren zu wollen.

5. Mathematische Grundlagen eines asymmetrischen Verschlüsselungsalgorithmus

5.1. Modulare Algebra[5]

Da man, wie oben erläutert, in der Kryptologie gerne schwer lösbare Probleme verwendet, die Rechen- und Speicherkapazitäten heutiger Computer jedoch beschränkt sind, nimmt man gerne Probleme, die sich in der modularen Arithmetik ergeben. Das hat zugleich den Vorteil, daß man die Länge der Zwischenergebnisse drastisch reduzieren kann. Wenn jeder Rechenschritt mit einem Modul der Länge k Bit ausgeführt wird, so wird es kein Zwischenergebnis mit einer Länge von mehr als 2k-1 Bit geben. So kann man auch mit Zahlen rechnen, die 200 und mehr Stellen haben, ohne die Benutzer des Kryptosystemes vor eine ungebührliche Geduldsprobe zu stellen. Die modulare Arithmetik gehorcht den selben

[5] nach [2], Seite 278ff.

Gesetzen wie die uns bekannte, reelle Algebra: Sie ist kommutativ, assoziativ und distributiv. Ferner hat sie den enormen Vorteil, daß es egal ist, ob man zuerst die Rechnung ganz ausführt oder das Ergebnis erst am Ende modular reduziert. Dadurch lassen sich, wie oben erwähnt, die Länge der Zwischenergebnisse begrenzen und zweitens auch enorm Zeit sparen.

5.2. Modulare Kongruenz[6]

Die Zahl 23, modular reduziert mit zwölf, ergibt elf. Man sagt dann, die Zahl 23 ist, bezüglich der modularen Reduktion mit zwölf kongruent zur Zahl elf. Auch die Zahl 35 ist kongruent zur Zahl elf: 35 MOD 12 = 11. Man schriebt diese Kongruenz als: $23 \equiv 11$ (MOD 12). oder allgemein: $a \equiv b$ (MOD n). Das Zeichen „≡“ bedeutet dabei: „ist kongruent“. Die Zahl b heißt das Residuum, und die Zahlen von 0 bis n –1 heißen die vollständige Residuenmenge modulo n.

5.3. Das Problem des modularen Inversen[7]

Der Kehrwert einer Zahl ist im allgemeinen einfach zu berechnen. Der Kehrwert k zu einer gegebenen Zahl a ist die Zahl, für die die Gleichung a * k = 1 erfüllt ist. In der modularen Arithmetik ist das Problem komplizierter. Dort ergibt sich die Zahl k zu 1 = (a * k) MOD n oder, in der Schreibweise der Kongruenzen: $a^{-1} \equiv k$ (MOD n). Die Berechnung des modularen Kehrwertes ist dabei nicht immer einfach, manchmal gibt es einfach keine passenden Zahlen. Die Berechnung dieser Werte werde ich später erläutern, hier soll zunächst einmal ein Überblick über sämtliche, mit der asymmetrischen Verschlüsselung zusammenhängenden Probleme gegeben werden.

5.4. Primzahlen[8]

Wie im Lauf dieser Arbeit noch deutlich werden wird, spielen Primzahlen eine große Rolle in der Kryptologie. Deswegen will ich im folgenden auf zwei mathematische Sätze eingehen, die helfen, große Primzahlen zu identifizieren.

---

[6] Quelle wie bei 5.1
[7] Quelle wie bei 5.2.
[8] nach [4], Kapitel 5

5.5. Der Satz von Fermat[9]

Der kleine Satz von Fermat ist eine Sonderform der Eulerschen – Phifunktion, auf die ich später noch eingehen werde. Der Satz von Fermat stellt für eine gegeben Primzahl p und eine natürliche Zahl a folgende Beziehung auf:

$$p \mid a^p - a$$

Leider jedoch gilt diese Beziehung nicht umgekehrt, die Mathematik gäbe uns so einen hervorragenden Primzahltest an die Hand. Da der Fermatsche Satz für die Bestimmung von Primzahlen extrem wichtig ist, werde ich ihn hier zunächst einmal beweisen, und zwar sowohl durch Induktion als auch auf kombinatorischem Wege:

Zunächst einmal gilt der Satz für a = 1, da $p \mid 1^p - 1$ klar ersichtlich ist. Dann nehme ich an, daß der Satz für eine natürliche Zahl a gilt, d.h.

(1) $p \mid a^p - 1$

Wenn dem so ist, muß aber auch gelten:

(2) $$p \mid (a+1)^p - (a+1)$$

Dabei wurde auf der linken Seite lediglich statt a a+1 gewählt und das Ergebnis auf der rechten Seite mittels Binomialkoeffizienten geschrieben. An der eigentlichen Formel wurden noch keinerlei Umstellungen vorgenommen. Des weiteren gilt:

$$p \left| \binom{p}{i}, 1 \le i \le p-1 \right.$$

[9] nach [4], Anhang 1

Vergleicht man diese Zeile mit der Umformung von (2), so erkennt man, daß p jeden Summanden auf der rechten Seite teilt. So kommt man zu dem Schluß, daß

$$p \mid (a+1)^p - (a+1)$$

ebenfalls richtig sein muß. Mit anderen Worten ist der Fermatsche Satz damit für alle a bewiesen. Wesentlich interessanter ist meiner Meinung jedoch der Beweis, der sich kombinatorischer Mittel bedient:

Man nehme Perlen in a verschiedenen Farben. Aus diesen werden Perlenschnüre mit jeweils p Perlen gelegt. Es entstehen a^p verschiedene Perlenschnüre. Von diesen sind a einfarbig. Legt man diese weg, so verbleiben a^p-a mehrfarbige Schnüre. verbindet man die beiden Enden jeder Schnur, erhält man „Halsketten“. Schnüre, bei denen die Perlen lediglich in ihrer Position mutiert sind, seien unterscheidbar. Da p eine Primzahl ist, gibt es p verschiedene Permutationen von p mehrfarbigen Perlen. Die Anzahl unterscheidbarer Halsketten ist daher

$$\frac{a^p - a}{p}$$

Aufgrund der Tatsache, daß es sich um „Halsketten“ handelt, ist dieses Ergebnis eine ganze Zahl. Daraus folgt dann:

$$p \mid a^p - a$$

Ist a nicht durch p teilbar, so folgt aus

$$a^p - a = a(a^{p-1} - 1)$$

zumindest, daß die Klammer durch p teilbar ist. Allerdings ist dann der ggT von a und p eins, also ist a zu eins modulo p kongruent. Womit der kombinatorische Beweis vollendet wäre.

5.6. Die Eulersche – Phifunktion und das Eulersche Kriterium[10]

Die Eulersche Phifunktion für eine beliebige Zahl p, bezeichnet mit φ(p), ist die Menge aller zu p teilerfremden Zahlen. Die Bedeutung der Phifunktion soll hier nicht näher untersucht werden, es soll nur um φ(pq) gehen, wobei p und q zwei Primzahlen seien sollen. Es gibt pq-1 Zahlen, die kleiner als pq sind. Es gibt q-1 bzw. p-1 Zahlen, die zu der Zahl nicht teilerfremd sind, nämlich die q-1 Vielfachen von p und die p-1 Vielfachen von q. Also ergibt sich:

$$\varphi(pq) = pq - 1 - (q-1) - (p-1) = pq - q - p + 1 = (p-1)(q-1)$$

Das Eulersche Kriterium nun sagt, daß für zwei natürliche Zahlen m und n, die zueinander teilerfremd sind, die Beziehung

$$m^{\varphi(n)} \equiv 1 (\mathrm{mod}\, n)$$

gilt, oder aber auch, für den Fall daß n eine aus zwei Primzahlen p und q zusammengesetzte Zahl ist

$$m^{(p-1)(q-1)} \equiv 1 (\mathrm{mod}\, pq)$$

Einen Beweis hierzu findet man in [3].

5.7. Der Euklidische und der erweiterte Euklidische Algorithmus[11]

Das ursprüngliche Ziel des euklidischen Algorithmus ist es, ggT(a,b), also den größten gemeinsamen Teiler zweier Zahlen auszurechnen. Bei großen Zahlen, die unmöglich in

[10] nach [1], Seite 123 ff.
[11] nach [2], Seite 123ff.

Primfaktoren zu zerlegen sind (Zahlen um die 250 Stellen gelten als unangreifbar, was die Zerlegung in Primfaktoren angeht), ist dieser Algorithmus die einzige Möglichkeit, den ggT zu bestimmen.
Man folgt dabei einem einfachen Schema. Gegeben seien zwei Zahlen a und b, mit a > b. Dann gilt einfach:

$r_0=a$

$r_1=b$

$r_0=q_1*r_1+r_2$ $\quad 0<r_2<r_1$

$r_1=q_2*r_2+r_3$ $\quad 0<r_3<r_2$

...

$r_{i-1}=q_i*r_i+r_{i+1}$ $\quad 0<r_{i+1}<r_i$

...

$r_{n-1}=q_{n-1}*r_{n-1}+r_n$ $\quad 0<r_n<r_{n-1}$

$r_{n-1}=q_n*r_n$

Die so gefundene Zahl r_n ist der größte gemeinsame Teiler von a und b. Daß er sie teilt, ist klar, da ein beliebiges r_p mit $p<n$ stets seine Vorgänger teilt r_{p-1} teilt. Es muß hier nicht bewiesen werden, daß der Algorithmus wirklich den größten Teiler liefert: Wir benötigen den Algorithmus lediglich, um festzustellen, ob zwei natürliche Zahlen teilerfremd zueinander sind. Wenn zwei Zahlen 1 als größten Teiler haben, wird der Algorithmus den Wert 1 liefern. Liefert er nicht den Wert 1, so sind die Zahlen nicht teilerfremd.
Weiter oben wurde bereits das Problem des modularen Inversen angesprochen. Gesucht wird eine Zahl a, so daß gilt: a*x sei kongruent zu 1 modulo b. Dies läßt sich auch schreiben als: ax+by=1. Setzen wir nun statt eins die Variable d, so handelt es sich um die Vielfachsummendarstellung des ggT's von a und b, und die Lösung ist gleichzeitig die Lösung der diophantischen Gleichung. Um dies zu beweisen, sei hier noch einmal die Gleichungskette des euklidischen Algorithmus untersucht:

$$
\begin{aligned}
&d = r_n = r_{n-2} - q_{n-1} \cdot r_{n-1} = r_{n-2} + (-q_{n-1}) r_{n-1} = \\
&= r_{n-2} - q_{n-1}(r_{n-3} - q_{n-2} \cdot r_{n-2}) = (-q_{n-1}) \cdot r_{n-3} + [1 + q_{n-2} \cdot q_n - 1] * r_{n-2} = \\
&[1 + q_{n-2} * q_{n-1}](r_{n-4} - q_{n-3} \cdot r_{n-3}) - q_{n-1} \cdot r_{n-3} = \\
&= [...] r_{n-4} + [...] r_{n-3} = \\
&= ... = \\
&= x \cdot r_0 + y \cdot r_1 = x \cdot a + y \cdot b
\end{aligned}
$$

Somit ist das Problem, ein modulares Inverses zu finden, gelöst.

5.8. Der Rabin – Monte –Carlo –Test[12]

Auf diesen Test wird nur kurz eingegangen. Er dient dazu, mit einer bestimmten Wahrscheinlichkeit sagen zu können, ob einen gegebene Zahl eine Primzahl ist. Das Verfahren dabei ist folgendes:
Es sei N ungerade und $N-1=2^s t$ (t ungerade). Man nennt N eine stake Pseudoprimzahl zur Basis b (starke b-PSP), wenn entweder

$$b^t \equiv 1 (\mathrm{mod}\, N)$$

oder es ein r,0<r<s gibt, so daß die Beziehung

$$b^{t 2^r} \equiv -1 (\mathrm{mod}\, N)$$

gilt. Eine Primzahl N ist eine starke b-PSP für jedes b von 1 bis N-1. Denn für eine Primzahl N gilt nach dem kleinen Fermat - Satz

$$b^{t 2^r} = b^{N-1} \equiv 1 (mod N)$$

und zwar für alle b bis N-1. Wenn b^t also nicht kongruent zu 1 modulo N ist, dann hat das letzte von 1 verschiedene Element der Folge $b^t, b^{2t}, b^{4t}, ..., b^{2^{s-1} t}$ (mod N) den Wert –1. Also ist N eine starke b-PSP.

[12] nach [4], Kapitel 5

Es sein dann N zerlegbar. Rabin zeigte, daß N dann für mindestens 75% aller b keine starke b-PSP ist. Solche b heißen Zeugen für die Zerlegbarkeit von N. Damit ergibt sich folgender Rabin – Test:
Um zu prüfen, ob N eine Primzahl ist, wählt man zufällig k Zahlen aus zwischen 1 und N-1. Für jedes ausgewählte b prüft man, ob b eine starke b-PSP ist. Wenn nicht, so ist N sicher zerlegbar. Wenn schon, so ist N eine Primzahl, und die Fehlerwahrscheinlichkeit ist $<4^k$.
Nunmehr sind alle mathematischen Voraussetzungen gegeben, um ein asymmetrisches Kryptosystem zu betrachten.

6. Das RSA – Verfahren

6.1. Geschichte[13]

Die grundlegende Idee zu Public – Key –Algorithmen entwickelten im Jahre 1976 Whitfield Diffie und Martin Hellman sowie unabhängig davon Ralph Merkle. Den genannten Herren stand allerdings noch kein geeigneter Algorithmus zur Verfügung, um ihre Idee zu verwirklichen. Das änderte sich im Jahre 1979, als die Mathematiker Ron Rivest, Adi Shamir und Leonard Adleman ihr Verfahren vorstellten. Es beruht auf dem Problem des modularen Inversen und der Tatsache, daß die Zerlegung einer Zahl in Primfaktoren bisher ein Problem ist, daß als NP – vollständig gilt, d.h. für daß man noch keinen polynomialen Algorithmus kennt.

6.2. Schlüsselerzeugung[14]

Als erstes sucht man sich zwei Primzahlen p und q, und berechnet $\varphi(pq)$ zu (p-1)(q-1). Dann sucht man zwei Zahlen e und d, so daß $e*d \text{ MOD } \varphi(pq) = 1$ ist. e und pq sind dann der geheime, d und pq der öffentliche Schlüssel. Es liegt dabei nahe, eine der Zahlen e und d frei zu wählen – wie gezeigt wird, ist es besser, e frei zu wählen.

[13] nach [2], Seite 531 ff.
[14] nach [1], Seite 128f.

6.3. Ver- und Entschlüsseln mit RSA[15]

Zum Verschlüsseln ist es notwendig, die Nachricht in Zahlen m kleiner als n zu zerlegen. Da die Verschlüsselung meist sowieso auf einem Computer stattfindet, bietet es sich an, den ASCII – Code zu benutzen.
Aus der Nachricht m wird durch die Verschlüsselung die Chiffre c: $c = m^e \text{ MOD } pq$.
Der umgekehrte Weg ist der gleiche: $m' = c^d \text{ MOD } pq$.
Es sollte darauf hingewiesen werden, daß das RSA – Verfahren das asymmetrische Verfahren ist, das am einfachsten nachzuvollziehen bzw. zu implementieren ist.
Die Korrektheit der Ver- und Entschlüsselung soll im folgenden bewiesen werden.
Es soll also gelten, daß $m^{e*d} \text{ MOD } n = m$ ist, daß heißt das die Botschaft nach dem Entschlüsseln mit der ursprünglichen Botschaft übereinstimmt. Nun sind e und d so gewählt, daß gilt: $e*d \text{ MOD } \varphi(pq) = 1$. Dann gibt es ein ganzzahliges, nichtnegatives k so daß gilt: $e*d = 1 + k*\varphi(pq) = 1+k*(p-1)(q-1)$. Ist m nun ein Vielfaches von p, so teilt p nicht nur m sondern natürlich auch m^{e*d}. Also gilt: $(m^{e*d}-m) \text{ MOD } p = 0$. Mit $\varphi(p) = p-1$ gilt nach Euler:

$$m^{e*d} \text{ MOD } p = m^{1+k*\varphi(n)} \text{ MOD } p = m*m^{k\varphi(n)} \text{ MOD } p = m*m^{k*(q-1)(p-1)} \text{ MOD } p =$$
$$= m*(m^{p-1})^{k*(q-1)} \text{ MOD } p = m*1^{k*(q-1)} \text{ MOD } p = m \text{ MOD } p$$

Für jede natürlich Zahl m gilt des weiteren: $(m^{e*d} - m) \text{ MOD } q = m$. Der Beweis ist analog zur obigen Gleichungskette auszuführen. Also teilen p und q beide $m^{e*d} - m$. Da q ungleich p ist, und beide Zahlen Primzahlen sind, teilt auch ihr Produkt $m^{e*d} - m$. Berücksichtigt man nun wiederum, wie p und q gewählt wurden, ergibt sich die folgende Aussage: $m^{e*d} \text{ MOD } pq = m$, was zu zeigen war.

6.4. Einsatz von RSA in einem hybriden, öffentlichen Kryptosystem

In einem solchen Kryptosystem wären die öffentlichen Schlüssel aller Teilnehmer in einer öffentlich zugänglichen Liste gespeichert. Die Schlüssel würden von einer öffentlichen Vergabestelle verteilt. Die Verschlüsselung selbst würde mit einem zufälligen Einweg – Schlüssel erfolgen. Der Schlüsselwürde an die Nachricht angehängt und mit dem öffentlichen Schlüssel des Empfängers codiert. Ein Beispiel hierfür ist das PGP – Programm: Öffentliche

[15] nach [1], Seite 129 - 133

Keyserver im Internet halten die Schlüssel bereit, und die Verschlüsselung erfolgt hybride mittels RSA und IDEA.

7. Schlußwort

Die Möglichkeiten, Botschaften geheim zu versenden, sind enorm. Jeder Teilnehmer eines Informatikkurses kann selbst Programme schreiben, die seine Botschaften so verschlüsseln, daß sie nicht mehr zu knacken sind. Angesichts dieser bereits offensichtlichen Verhältnisse wirkt die geplante Kryptoregulierung der Bundesregierung wie blanker Hohn. Sie würde eine Kriminalisierung derjenigen bewirken, die sich ihr Recht auf eine Privatssphäre und freien, unabhörbaren Meinungsaustausch nicht nehmen lassen wollten. Hier nun ist die Mathematik gefordert: Sie muß Verfahren bereitstellen, die dem Statt ein dekodieren von Botschaften ermöglichen – und NUR ihm. Wenn en solches Verfahren gefunden würde, wären alle Diskussionen um Schlüsselhinterlegung und Beschränkung der Exporte von Sicherheitssoftware völlig überflüssig.

8. Literaturverzeichnis

[1] Albrecht Beutelspacher: Kryptologie, Friedrich Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig /Wiesbaden, 1994

[2] Elmar Warken: Angewandte Kryptologie, Addison – Wesley GmbH, Bonn, 1996

[3] Harald Scheid: Zahlentheorie, BI Wissenschafts – Verlag, Mannheim; Wien; Zürich, 1991

[4] A. Engels: Datenschutz durch Chiffrieren: Mathematische und algorithmische Aspekte, O.V., O.O, O.J.

[5] M. Bossert: Kryptologie, Universität Ulm – Abteilung Informationstechnik, O.O., O.J