

# Strukturvergleich zwischen Klartext und Geheimtext verschlüsselter Dateien

Bedeutung der Längenverhältnisse

(Ernst Erich Schnoor)

- A. Geheimtext als Abbildung des Klartextes
- B. Strukturen von Klartext und Geheimtext
  - 1 Statistische Gesetzmäßigkeiten der Sprache
    - 1.1 Wiederholungsmuster und Häufigkeiten
    - 1.2 Invarianz von "Kappa" und "Chi"
    - 1.3 Unizitätslänge, Redundanz und Entropie
  - 2 Verschlüsselungstechniken
    - 2.1 Klassische Kryptographie
      - 2.1.1 Substitution
      - 2.1.2 Transposition und Permutation
      - 2.1.3 Codebücher
      - 2.1.4 Cäsar, Vigenere und Vernam
    - 2.2 Elektronische Kryptographie
      - 2.2.1 symmetrische und asymmetrische Verfahren
      - 2.2.2 Strom- und Blockchiffrierungen
      - 2.2.3 XOR-Verknüpfungen
      - 2.2.4 Feistel-Netzwerke
      - 2.2.5 Betriebsarten: ECB, CBC, CFB und OFB
      - 2.2.6 DES, IDEA, AES, RC4, RSA, ElGamal und andere
      - 2.2.7 Encoding Base 64
- C Angriffstechniken
  - 1 Koinzidenzanalyse (Kappa)
  - 2 Parallelstellensuche (Kasiski)
  - 3 Ciphertext-Only-Angriffe (brute force)
  - 4 Known-Plaintext-Angriffe
  - 5 Chosen-Plaintext-Angriffe
  - 6 Differenzielle und lineare Kryptanalyse
  - 7 Sonstige Angriffsszenarien
- D Schlussfolgerungen
  - 1 Folgen fehlender Längenkongruenz
  - 2 Abschließende Bemerkungen
- E Literaturhinweise

## A. Geheimtext als Abbildung des Klartextes

Die Kryptographie hat die Aufgabe, einen Klartext in der Weise in einen Geheimtext umzuwandeln (Chiffrierung), dass nur der berufene Empfänger - und kein anderer - in der Lage ist, die verschlüsselte Information zu lesen. Dazu sind eine **funktionale Verbindung** zwischen Klartext und Geheimtext (Algorithmus) und ein Parameter (Schlüssel) erforderlich, die die Funktion zum Arbeiten bringen. Im funktionalen Fall ist die Chiffrierung deterministisch und insoweit eine eindeutige Abbildung des Klartextbereichs in den Geheimtextbereich [#1].

Die funktionale Verbindung ist allerdings auch das Ziel von Angriffen, die den Schlüssel suchen oder den Klartext auf andere Art und Weise herausfinden möchten (Kryptanalyse). Hier gibt es bisher und heute viele Versuche und Methoden, die zum Teil erfolgreich sind, aber möglichst ohne Erfolg bleiben sollen.

In der historischen Entwicklung waren die Kryptographen bemüht, für jeden Buchstaben ein bestimmtes Geheimzeichen festzulegen. Dadurch ergeben sich so viele Geheimzeichen wie Klartextbuchstaben vorhanden sind. Der Geheimtext wird also genau so lang wie der Klartext. (in **Zeicheneinheiten**) Das scheint auch heute im Zeitalter der elektronischen Kryptographie noch nicht viel anders zu sein. Viele Beteiligte gehen implizit immer noch davon aus, dass Klartext und Geheimtext gleich lang sind, ja sogar gleich lang sein müssen: [#2] **Längenkongruenz** .

Bei gleicher Länge muss für jeden Klartextbuchstaben auch ein bestimmtes Geheimzeichen in irgend einer Form existieren oder jedenfalls durch die funktionale Verbindung zugeordnet werden können. Dann stellt sich die weitere Frage, ob das einem bestimmten Klartextbuchstaben zuzuordnende Geheimzeichen auch an der gleichen Position wie im Klartext steht (**Strukturanalyse**).

In der Fachliteratur sind diese Aspekte kaum oder gar nicht behandelt. Wenn sie überhaupt angesprochen werden, dann allenfalls mit der Begründung, der Geheimtext dürfe nicht länger sein als der Klartext, da er ja an der gleichen Stelle wieder abgelegt werden solle [#3].

Im Folgenden wird nun untersucht, inwieweit die Längen und Strukturen von Klartext und Geheimtext für kryptographische Verfahren von Bedeutung sind. Dabei bleiben der Einfluss von Headern, Blendern, Checksummen, Kontrollsequenzen, Zeitstempel und in der klassischen Kryptographie auch noch Leerzeichen außer Betracht. Die jeweiligen Ergebnisse sind im kleinen Kästchen am Ende des Absatzes gekennzeichnet.



## B. Strukturen von Klartext und Geheimtext

Bekanntlich fing die Verschlüsselung damit an, dass ein Buchstabe an seiner Stelle im Klartext durch ein Geheimzeichen ersetzt wurde [#4]. Mit zunehmender Durchdringung der Materie und erweiterten technischen Möglichkeiten - auf beiden Seiten: Kryptographie und Kryptanalyse - wurde diese einfache Zuordnung durch nicht so einfach zu analysierende Strukturen erweitert [#5]. Die Länge der Geheimzeichensequenz stimmt nicht immer mit der entsprechenden Klartextlänge überein und das verschlüsselte Geheimzeichen steht vielfach an einer anderen Stelle als im Klartext. Diese Möglichkeiten stoßen allerdings auf ein besonderes Phänomen: die **Invarianz** vieler Eigenschaften der Sprache [#6].

### 1 Statistische Gesetzmäßigkeiten der Sprache

Jede Sprache enthält ein schwer ausrottbares inneres Gerüst von Gesetzmäßigkeiten [#7]. Infolge der funktionalen Verbindung zwischen Klartext und Geheimtext kommen die Eigenschaften der Sprache in irgend einer Form auch im Chiffretext zum Ausdruck (direkt oder indirekt). Sie müssen nur gefunden werden oder sie sind so versteckt, dass sie nicht zu finden sind.

#### 1.1 Wiederholungsmuster und Häufigkeiten

Die am meisten ins Auge fallenden Eigenschaften sind Wiederholungen und Häufigkeiten der Zeichen. Für monoalphabetische Substitutionen gilt: "Wiederholungsmuster der Einzelzeichen innerhalb des Textes bleiben erhalten" **Invarianzsatz 1** [#8]. Bei der sogenannten "negativen Mustersuche" in polyalphabetischen Chiffrierungen gilt das Gleiche. Diese Feststellungen erstrecken sich ausdrücklich auf **Einzelzeichen** , so dass für jedes Klartextzeichen auch ein

bestimmtes Geheimtextzeichen vorhanden ist, also Klartext und Geheimtext gleich lang sind und die Struktur unverändert bleibt.



Für alle Transpositionen stellt der **Invariansatz 2** fest: "Häufigkeiten der Einzelzeichen innerhalb des Textes bleiben erhalten" [#9]. Der Hinweis auf die Einzelzeichen führt hier ebenfalls zu der Schlussfolgerung, dass zwischen Klartext und Geheimtext eine Längenkongruenz besteht. Die Analyse und Darstellung von Einzelzeichen in "Häufigkeitsgebirgen, Cliques und Partitionen" [#10] wären auch nicht sinnvoll, wenn Klartext und Geheimtext ungleiche Längen hätten. Allerdings die Strukturen können sich verändern.



## 1.2 Invarianz von "Kappa" und "Chi"

Die relative Häufigkeit von zeichenweisen übereinstimmungen bei übereinander gelegten Texten (Zeichenkoinzidenz) bezeichnet man bekanntlich als **Kappa** der beiden Texte [#11]. Um das KAPPA zur Analyse chiffrierter Texte verwenden zu können, werden in den **Variansätzen 5** und **6** ausdrücklich **gleich lange** Texte vorausgesetzt [#12]. Gleiches gilt auch für die weiteren Theoreme **Chi** und **Sigma** [#12]. Eine Übereinstimmung der Strukturen liegt allerdings nicht vor, da diese Analyseschritte ja gerade die Gesetzmäßigkeiten herausfinden sollen.



## 1.3 Unizitätslänge, Redundanz und Entropie

Nach **Shannon** [#14] ist die **Unizitätslänge** definiert als "Näherung für die Menge verschlüsselten Texts, bei der die Summe aus echter Information (Entropie) im zugehörigen Klartext und der Entropie des Chiffrierschlüssels gleich der Anzahl der im Chiffretext benutzten Bits ist" [#15]. In symmetrischen Kryptosystemen wird die Unizitätslänge auch definiert als die Entropie des Kryptosystems dividiert durch die Redundanz der Sprache [#15a]. Klartext und Geheimtext werden zueinander in Beziehung gesetzt, so dass sich die Frage nach der Längenkongruenz stellt. Die Unizitätslänge wird zwar auf eine bestimmte Anzahl von Zeichen im Klartext bezogen [#16], da aber die Unizitätslänge offensichtlich sich nur auf die Beurteilung der Effektivität eines "brute force" Angriffs bezieht, scheint hier ein Vergleich zwischen Klartext- und Geheimtextzeichen nicht angezeigt zu sein [#17].



Die Informationsgrößen **Redundanz** und **Entropie** beziehen sich stets auf eine bestimmte Sprache. Die Redundanz in einer Klartextnachricht wird am besten durch "Konfusion" und "Diffusion" verborgen. Das vereitelt die Suche nach Gesetzmäßigkeiten und statistischen Mustern [#18]. Die Entropie eines Klartextes zählt die Anzahl der Klartextbits, die man wiederherstellen muss, um eine verschlüsselte Nachricht zu entziffern [#19]. Allgemein wird die Entropie jedoch auch als Maß für die Größe des Schlüsselraums eines Kryptosystems verwendet.



Insgesamt bleibt festzustellen, dass bei Unizitätslänge, Redundanz und Entropie die Längen und Strukturen von Klartext und Geheimtext zwar eine Rolle spielen, aber ein Einzelbezug von bestimmten Klartextzeichen auf bestimmte Geheimtextzeichen nicht von Bedeutung ist.

## 2. Verschlüsselungstechniken

Im Laufe der Zeit haben sich die Verschlüsselungstechniken grundlegend verändert. Bis zum Arbeiten mit Computern waren die Algorithmen **zeichenorientiert** (klassische Kryptographie). Gegenstand der Verschlüsselung war das einzelne Zeichen. Mit dem Einsatz elektronischer Hilfsmittel treten dann einzelne Bits an die Stelle der Zeichen. Die aktuellen Algorithmen sind heute fast ausschließlich **bitorientiert**, auch wenn daneben noch einige **Bytes basierte** Verfahren verwendet werden [#20].

### 2.1 Klassische Kryptographie

#### 2.1.1 Substitution

Bei dem Chiffrierschritt **Substitution** wird jedes Zeichen des Klartextes durch ein anderes Zeichen (Geheimzeichen) im Chiffretext ersetzt. Bei den vier Arten der Substitution (einfache, homophone, polygraphische und polyalphabetische Substitution) bleiben die Positionen der betreffenden Zeichen im Klartext und im Geheimtext gleich, auch wenn für die einzelnen Substitutionen unterschiedliche Ersetzungen vorgenommen werden [#21]. In allen Verfahren, die Substitutionen verwenden, sind daher die Längen und die Struktur von Klartext und Geheimtext grundsätzlich gleich.



#### 2.1.2 Transposition und Permutation

Im Vergleich zur Substitution wird bei der **Transposition** zusätzlich noch die Position des Klartextzeichens im Geheimtext verändert. Die Buchstaben des Klartextes werden umstrukturiert, insoweit bleibt die Längenkongruenz zwischen Klartext und Geheimtext bestehen [#22]. Wenn aber die Zeichen des Klartextes von einem Bitsystem - z.B. zur Basis 8 - in ein anderes Bitsystem - z.B. zur Basis 7 - umgewandelt werden (Bit Konversion), wird die Längenkongruenz durchbrochen.



Bei der **Permutation** werden die Buchstaben des Klartextes in der Form durcheinander gewürfelt, dass im Geheimtext kein Zeichen mehr an der gleichen Position wie im Klartext verbleibt [#23]. Da die Anzahl der Zeichen sich nicht ändert, bleiben auch die Längen beider Texte gleich, nur die Struktur verändert sich.



#### 2.1.3 Codebücher

Codierschritte aus **Codebüchern** zeigen gegenüber den bisherigen Situationen eine wesentliche Ausnahme: es gibt keinen funktionalen Zusammenhang, es sei denn, man betrachtet das Lesen des Codebuchs und das Aufschreiben der Zeichen als "Funktion" [#24]. Chiffriertabellen in einem Codebuch lassen sich nach den unterschiedlichsten Gesichtspunkten und Formulierungen zusammenstellen. Der Phantasie des Kryptographen sind hier keine Grenzen gesetzt mit der Folge, dass zwischen Klartext und Geheimtext weder Längenkongruenz noch vergleichbare Strukturen bestehen.



## 2.1.4 Cäsar, Vigenere und Vernam

Die **Cäsar-Chiffre** ist die einfachste Form einer Substitution [#25]. Jedes Klartextzeichen wird an seiner Stelle in ein Geheimzeichen umgewandelt. Daher sind grundsätzlich sowohl Längenkongruenz als auch Strukturgleichheit gegeben.



Das Verfahren nach **Blaise de Vigenere** ist eine polyalphabetische Chiffre. Es setzt sich im Prinzip aus mehreren Cäsar-Chiffren zusammen, nämlich genau so vielen wie der Schlüssel Buchstaben hat [#26]. Ein Vigenere-Chiffrierschritt bewirkt eine einfache lineare Substitution [#27]. Auch wenn das Vigenere-Verfahren mit polyalphabetischen Methoden arbeitet bleiben dennoch gleiche Längen von Klartext und Chiffretext erhalten.



Ein **Vernam-Chiffrierschritt** stellt ein bitweises Vigenere-Verfahren dar, wird allerdings in Binärverschlüsselung durchgeführt [#28]. Das Verfahren ist besonders interessant durch seine Erweiterungsmöglichkeit zum **One-Time-Pad** [#29]. Allerdings muss der Schlüssel in diesem Fall auch so lang sein, wie der Klartext. Es besteht daher Längenkongruenz zwischen Klartext, Schlüssel und Geheimtext. Die XOR-Verknüpfung hat auch eine gleiche Struktur zur Folge.



## 2.2 Elektronische Kryptographie

Der Übergang von zeichenweise arbeitenden Verfahren zur elektronischen Verschlüsselung ist bei einigen Verfahren fließend, beispielsweise beim Vernam-Verfahren. Die Klartextzeichen werden einfach durch einzelne Bits ersetzt und gewöhnlich bilden 8 Bits ein Byte (als klassisches Zeichen). Im Grundsatz bleiben viele Verfahren gleich.

### 2.2.1 Symmetrische und asymmetrische Verfahren

Die Unterscheidung in **symmetrische** und **asymmetrische** Algorithmen betrifft nur die anzuwendenden Schlüssel: Während bei symmetrischen Systemen ein gemeinsamer Schlüssel für Sender und Empfänger verwendet wird sind in der Publik-Key-Kryptographie zwei Schlüssel erforderlich - ein öffentlicher Schlüssel und ein geheimer privater Schlüssel. Das Verhältnis von Klartext zum Geheimtext wird durch diese Unterscheidung jedoch nicht beeinflusst. Alle Chiffrierschritte können gleichermaßen angewendet werden. Eine konkrete Feststellung ist damit an dieser Stelle noch nicht zu treffen. Es kommt auf den einzelnen Fall an.

### 2.2.2 Strom- und Blockchiffren

Stromchiffrierungen und Blockchiffrierungen sind zwei unterscheidbare Kategorien symmetrischer Verfahren [#30]. **Stromchiffrierungen** bearbeiten seriell immer ein Bit (oder auch ein Byte) von Klartext und Chiffretext. Die Stromchiffrierung liefert für das gleiche Klartextbit (oder Klartextbyte) bei jeder Verschlüsselung ein anderes Bit (oder Byte) [#31]. Daraus folgen eine absolute Längenkongruenz und gleiche Strukturen zwischen Klartext und Chiffretext.



**Blockchiffrierungen** bearbeiten Klartext und Chiffretext in Blöcken von meist 64 Bit Länge, die aber auch kürzer oder länger sein können. Bei Verwendung des gleichen Schlüssels folgt aus einem bestimmten Klartextblock immer der gleiche Chiffretextblock [#32]. Wenn diese Feststellung auch auf gleiche Längen zwischen Klartextblöcken und Chiffretextblöcken schließen lässt, so hängt es

unter Umständen noch vom angewandten Betriebsmodus ab, wie die Längen sich tatsächlich verhalten. In manchen Fällen - beispielsweise beim **Padding** - ist es jedoch unbedingt notwendig dass der Chiffretext die gleiche Länge hat wie der Klartext [#33]. Die Strukturanalyse bei Blockchiffrierungen wird in den meisten Fällen durch "Konfusion" und "Diffusion" erschwert, wenn nicht sogar unmöglich gemacht [#34].



### 2.2.3 XOR-Verknüpfungen

Bits in gleicher Position werden per XOR, d.h. exklusivem OR, verknüpft [#35]. Als Verschlüsselungsschritt stellt die **XOR-Verknüpfung** lediglich eine polyalphabetische "Vigenere-Chiffrierung" dar [#36]. Die Anwendung führt zur Längenkongruenz und zur Strukturenidentität.



### 2.2.4 Feistel-Netzwerke

Die meisten Blockchiffrierungen sind **Feistel-Netzwerke** [#37]. Entsprechend sind die Längen der Eingabe- und Ausgabeblöcke gleich, also eine Längenkongruenz ist gegeben. Mit dem kombinierten Einsatz von "Konfusion" und "Diffusion" (**Produktchiffrierung**) wird ein Strukturvergleich verhindert [#38].



### 2.2.5 Betriebsarten: ECB, CBC, CFB und OFB

Die genannten Betriebsarten sind für alle Blockchiffrierungen anwendbar. ECB und CBC arbeiten direkt als Blockchiffrierung während CFB und OFB den Blockalgorithmus nur verwenden, um eine Stromchiffrierung zu definieren [#39]. Die Verfahren wandeln die Klartextblöcke nacheinander um in zugehörige Geheimtextblöcke. Mit Initialisierungsvektoren kommen unterschiedliche Wirkungen in die Abläufe. Allen gemeinsam aber bleibt die Längenkongruenz zwischen Klartext und Geheimtext. Strukturanalysen spielen bei den Betriebsmodi keine erkenntnistheoretische Rolle.



### 2.2.6 DES, IDEA, AES, RC4, RSA, ElGamal und andere

#### **DES** (Data Encryption Standard)

DES ist ein Produktalgorithmus, speziell ein Feistel-Netzwerk [#40]. Der Chiffretext ist nicht länger als der Klartext [#41]. Die Längenkongruenz ist gegeben. Ein Strukturvergleich zwischen Klartext und Chiffretext ist allerdings nicht möglich.



#### **IDEA** (International Data Encryption Algorithm)

Der Algorithmus verschlüsselt in acht Runden, ist also eine Produktchiffrierung, jedoch kein Feistel-Netzwerk [#42]. Die Eingangs-Teilblöcke ( $x_1 - x_4$ ) haben die gleiche Länge wie die Ausgangs-Teilblöcke ( $y_1 - y_4$ ) [#43], so dass eine Längenkongruenz vorliegt. Ein Strukturvergleich wird jedoch durch "Konfusion" vermieden.



## AES (Advanced Encryption Standard)

AES (Rijndael) ist eine symmetrische Blockchiffrierung. Die Schlüssel können mit 128, 192 oder 256 Bits festgelegt werden. Eingabe- und Ausgabeblocke bestehen aus 128 Bit Sequenzen. In jedem Fall sind sie gleich lang. Damit besteht eine Längenkongruenz. Infolge inverser Spaltentechnik werden die Strukturen vermischt. Eine Strukturzuordnung ist daher nicht gegeben [#44].



## RC4 (Rivest Cipher Nr.4) [#45]

RC4 ist eine Stromchiffrierung auf Basis von Verarbeitung ganzer Zeichen (Bytes) [#46]. Da jedes Klartextzeichen im Strom sofort verschlüsselt wird, sind Eingabestrom und Ausgabestrom in der Länge identisch. Die Struktur stimmt ebenfalls überein, da jedes Geheimzeichen an der selben Stelle steht wie im Eingabestrom.



## RSA (Rivest, Shamir und Adleman)

RSA ist der beliebteste Algorithmus mit geheimem privaten und öffentlichem Schlüssel (asymmetrisches Verfahren) [#47]. Klartext, Chiffretext und Schlüssel sollte man sich nicht als Bitfolgen, sondern als natürliche Zahlen vorstellen. Für die Analyse macht das aber keinen Unterschied [#48]. Der Kern des Verfahrens liegt in der mathematischen Behandlung der Schlüsselmechanik (öffentlicher und privater Schlüssel).

Die Chiffrierung arbeitet mit Klartextblöcken (N -1 Bit) in Abhängigkeit von der Länge des Schlüssels (N Bit). Der Geheimtextblock hat dann ebenfalls die Länge (N Bit). Durch entsprechendes Auffüllen (Padding) lassen sich Geheimtext- und Klartextblöcke auf gleiche Länge bringen [#49]. Die Längenkongruenz liegt somit vor. Ein Strukturvergleich wird durch die Blocktechnik verhindert.



## ElGamal

Das Verfahren von **T.ElGamal** wird für digitale Signaturen und Verschlüsselungen verwendet. Das kryptographisch "harte Problem" liegt in der Schwierigkeit, diskrete Logarithmen über einen endlichen Körper zu berechnen [#57]. Für die Verschlüsselung steht die Feststellung: "Der Geheimtext ist doppelt so lang wie der Klartext" [#58]. Eine Übereinstimmung in der Länge ist daher nicht gegeben. Dementsprechend weichen auch die Strukturen voneinander ab.



Andere Verfahren, wie Blowfish, SAFER, FEAL, GOST und weitere stützen sich vor allem auf unterschiedliche Chiffriertechniken (Blockalgorithmen, S-Boxen) und nicht auf die Längenproblematik von Eingaben und Ausgaben. Klartextsequenzen und die entsprechenden Geheimtextsequenzen - verglichen in Bits - sind in fast allen Fällen gleich lang. Lediglich die Strukturen lassen keine speziellen Vergleiche zu.

### 2.2.7 Encoding Base 64

Beim Verfahren "**Coding Base 64**" werden 8-bit Sequenzen des Klartextes in eine Folge von **6-bit Sequenzen** umgewandelt, deren dezimale Werte als Indizes für ein Chiffre-Alphabet von 64 Zeichen verwendet werden. Das Chiffre-Alphabet kann entweder **statisch** vorgegeben oder **dynamisch** in Verschlüsselungsrunden immer wieder neu generiert werden. Infolge des Verhältnisses von 6 Klartextzeichen für 8 Chiffretextzeichen ist keine Übereinstimmung in der Länge gegeben. Strukturidentität liegt ebenfalls nicht vor, da ein Klartextzeichen keinem bestimmten Chiffretext-Zeichen zugeordnet werden kann.



## C. Angriffstechniken

Mit Angriffen auf Geheimtexte sollen möglichst der Schlüssel und/oder der Klartext herausgefunden werden. Da in den meisten Fällen anfangs nur der Geheimtext vorliegt, spielen die Länge von Geheim- und Klartext schon eine wichtige Rolle. Dabei sind sowohl klassische als auch moderne Analysetechniken in Betracht zu ziehen.

### 1 Koinzidenzanalyse (Kappa)

Die **Koinzidenzanalyse** von Geheimtexten dient vor allem der Suche nach der Länge des verwendeten Schlüssels und der resultierenden Periodenlänge. Es gilt: "Das KAPPA des um N Positionen gegen sich selbst verschobenen Geheimtextes ist gleich dem analog berechneten Kappa des Klartextes" [#50]. Da hier KAPPA gleichzeitig auf Klartext und Geheimtext bezogen wird, muss auch eine Längenkongruenz unterstellt werden. Die Struktur beider Texte spielt keine Rolle, sonst wäre die Koinzidenzanalyse ja gar nicht erforderlich.



### 2 Parallelstellensuche (Kasiski)

Die Suche nach parallelen Sequenzen (Multigramme) erstreckt sich im Prinzip nur auf Geheimtexte, die mit demselben Schlüssel chiffriert worden sind. Die **Parallelstellensuche** ist ein Verfahren, den Schlüssel zu finden, um dann den Klartext entschlüsseln zu können (Ciphertext-Only-Angriff). Ein Vergleich von Geheimtext und Klartext findet nicht statt, so dass auch deren Länge nicht von Bedeutung ist.



### 3 Ciphertext-Only-Angriffe (brute force)

Beim **Ciphertext-Only-Angriff** steht dem Angreifer nur der Geheimtext zur Verfügung. Der Klartext oder vorher noch der Schlüssel sollen ausschließlich aus dem Geheimtext gewonnen werden. Dabei ist es das Einfachste, aber auch das Schwierigste, den Schlüssel durch Ausprobieren aller Möglichkeiten zu finden (**brute force attack**). Die Längenkongruenz und die Vergleichbarkeit der Strukturen fallen hierbei nicht ins Gewicht.



### 4 Known-Plaintext-Angriffe

Bei einem **Known-Plaintext-Angriff** ist zusätzlich zum Geheimtext auch noch ein Teil des Klartextes bekannt. Im Falle einer Blockchiffrierung beispielsweise liegen ein Block Chiffretext und der entsprechende Klartextblock vor [#51]. Hier lassen sich bereits Beziehungen zwischen Klartext und Chiffretext erkennen, so dass von einer Längenkongruenz auszugehen ist. Eine Veränderung der Strukturen hat keinen Einfluss.



### 5 Chosen-Plaintext-Angriffe

Vom Angreifer wird ein bekannter Klartext unterschoben (**Chosen-Plaintext-Angriff**) in der Erwartung, er werde demnächst den chiffrierten Geheimtext, der vermutlich mit dem zu suchenden

Schlüssel verschlüsselt worden ist, abfangen können. Beim Vergleich von Klartext und Geheimtext besteht Längenkongruenz; Strukturvergleich ist aber nicht entscheidend.



## 6 Differenzielle und lineare Kryptanalyse

Bei der **differenziellen Kryptanalyse** betrachtet man gezielt gewisse Blöcke von Geheimtexten, nämlich solche, deren zugehörige Klartexte bestimmte Differenzen aufweisen [#52]. Dann werden diese Differenzen (veränderte Bits) nach erfolgter Verschlüsselung mit demselben Schlüssel wieder untersucht. Die Differenz entsteht "in einem Zahlkörper, der nur aus **gleichlangen** Folgen von Nullen und Einsen besteht" [#53]. Die differenzielle Kryptanalyse setzt insoweit eine Längenkongruenz von Klartext und Geheimtext voraus. Strukturvergleiche sind nicht gefragt.



Die **lineare Kryptanalyse** verknüpft einige Bits des Klartextes per XOR und dann Bits des Chiffretextes ebenfalls per XOR. Es entsteht dann ein einzelnes Bit, das der XOR-Verknüpfung einiger Bits des Schlüssels entspricht. Aus Klartext und zugehörigem Chiffretext wird nun versucht, die Werte der Schlüsselbits zu erraten [#54]. Diese Vorgänge können natürlich nur dann Erfolg haben, wenn zwischen Klartext und Chiffretext eine Längenkongruenz besteht. Strukturfragen werden nicht angesprochen.



## 7 Sonstige Angriffszenarien

Im kryptanalytischen Bemühen werden noch etliche Verfahren angewendet, deren Untersuchung aber zu weit führen würde. Als bemerkenswert ist sicherlich noch der **Angriff durch Einfügen** zu nennen, der insbesondere bei Stromchiffrierungen versucht wird [#55]. Der Angreifer benötigt Geheimtext und zumindest einige Bits (oder Bytes) Klartext. Längenkongruenz ist also erforderlich. Dagegen muss die Struktur der Zeichen nicht vergleichbar sein.



## D. Schlussfolgerungen

### 1 Folgen fehlender Längenkongruenz

Von den klassischen Angriffen setzt nur die **Koinzidenzanalyse** eine Längenkongruenz zwischen Klartext und Chiffretext voraus. Die **Parallelstellensuche** ist unabhängig von den Längen durchführbar.

Die im binären Bereich angesiedelten Angriffe – außer **brute force**, **ElGamal** und **Encoding Base 64** - erfordern alle übereinstimmende Klartext- und Geheimtextlängen (Längenkongruenz). Diese Erkenntnis ist besonders für das vom Autor entwickelte **CypherMatrix** Verfahren von Bedeutung [#56]. Dort wird die Längenkongruenz durch eine **Bit Konversion** ausgeschlossen (auf ein Klartextzeichen entfallen 8/7 Chiffretextzeichen), so dass alle vorstehend untersuchten Angriffe nicht angewendet werden können.

Auch für einen "**brute force**" Angriff lässt sich mathematisch ableiten, dass kein eindeutiges Ergebnis erzielt werden kann. Ausführliche Erläuterungen finden Sie im Artikel:

"Bit-Konversion im CypherMatrix Verfahren.

## 2 Abschließende Bemerkungen

In diesem Artikel werden eingetragene Warenzeichen, Handels- und Gebrauchsnamen verwendet. Auch wenn diese nicht ausdrücklich als solche gekennzeichnet sind, gelten die entsprechenden Schutzbestimmungen.

Vervielfältigungen und Übersetzungen dieses Artikels, auch auszugsweise, bedürfen der ausdrücklichen Genehmigung des Autors. Kritik, Anregungen, Verbesserungen und Korrekturen zu den Aussagen und Feststellungen sind jederzeit willkommen.

Ihre Nachricht erbitten wir per e-mail an:  
eschnoor@multi-matrix.de

München, im Februar 2004

### E. Literaturhinweise

- [#1] Bauer, Friedrich L., Entzifferte Geheimnisse - Methoden und Maximen der Kryptologie, Berlin Heidelberg New York, 1995, S. 27,
- [#2] Schmech, Klaus, Safer Net, Kryptografie im Internet und Intranet, Heidelberg 1998, S. 61,
- [#3] Schneier, Bruce, Angewandte Kryptographie (deutsche Ausgabe), Bonn ... 1996, S. 229,
- [#4] Wobst, Reinhard, Abenteuer Kryptologie, Bonn 1997, S. 24,
- [#5] Schmech, Klaus, a.a.O., S. 59,
- [#6] Bauer, Friedrich L., a.a.O., S. 186 ff.,
- [#7] Bauer, Friedrich L., a.a.O., S. 186,
- [#8] Bauer, Friedrich L., a.a.O., Sn. 186 ff.,
- [#9] Bauer, Friedrich L., a.a.O., S. 213,
- [#10] Bauer, Friedrich L., a.a.O., Sn. 220,
- [#11] Bauer, Friedrich L., a.a.O., S. 247,
- [#12] Bauer, Friedrich L., a.a.O., S. 249,
- [#13] Bauer, Friedrich L., a.a.O., Sn. 250 ff,
- [#14] Shannon, C.E., Communication Theory of Secrecy Systems, Bell System Technical Journal v.28, n.4, 1949, Sn. 379 ff.,
- [#15, 15a] Schneier, Bruce, a.a.O., S. 276,
- [#16] Bauer, Friedrich L., a.a.O., S. 180,
- [#16] Schneier, Bruce, a.a.O., S. 277,
- [#18] Schneier, Bruce, a.a.O., S. 274,
- [#19] Schneier, Bruce, a.a.O., S. 273,
- [#20] z.B. RC4 von Ron Rivest,
- [#21] Schneier, Bruce, a.a.O., S. 11,
- [#22] Schneier, Bruce, a.a.O., Sn. 13 und 276,
- [#23] Schmech, Klaus, a.a.O., S. 56,
- [#24] Bauer, Friedrich L., a.a.O., S. 28,
- [#25] Schmech, Klaus, a.a.O., S. 53,
- [#26] Schmech, Klaus, a.a.O., Sn. 57,
- [#27] Bauer, Friedrich L., a.a.O., S. 95,
- [#28] Wobst, Reinhard, a.a.O., S. 37 und Bauer, Friedrich L., a.a.O., S.108,
- [#29] Schneier, Bruce, a.a.O., S. 17,
- [#30] Schneier, Bruce, a.a.O., S. 4,

- [#31] [#32] Schneier, Bruce, a.a.O., S. 223,
  - [#33] Wobst, Reinhard, a.a.O., S. 168,
  - [#34] Schneier, Bruce, a.a.O., S. 400,
  - [#35] Wobst, Reinhard, a.a.O., S. 334,
  - [#36] Schneier, Bruce, a.a.O., S. 16,
  - [#37],[#38] Schneier, Bruce, a.a.O., Sn. 400 ff,
  - [#39] Wobst, Reinhard, a.a.O., S. 161,
  - [#40] Wobst, Reinhard, a.a.O., S. 115,
  - [#41] Schmeh, Klaus, a.a.O., S. 70,
  - [#42] Wobst, Reinhard, a.a.O., S. 183,
  - [#43] Wobst, Reinhard, a.a.O., S. 186,
  - [#44] Federal Information Processing Standards (FIPS PUB 197)  
Nov. 26,2001, Sn. 7, 13,
  - [#45] Schmeh, Klaus, a.a.O., S. 75,
  - [#46] Wobst, Reinhard, a.a.O., S. 199,
  - [#47] Schneier, Bruce, a.a.O., S. 20,
  - [#48] Schmeh, Klaus, a.a.O., S. 93,
  - [#49] Wobst, Reinhard, a.a.O., S. 148,
  - [#50] Wobst, Reinhard, a.a.O., S. 81,
  - [#51] Schneier, Bruce, a.a.O., S. 177,
  - [#52] Schneier, Bruce, a.a.O., S. 177,
  - [#53] Wobst, Reinhard, a.a.O., S. 125,
  - [#54] Schneier, Bruce, a.a.O., S. 338,
  - [#55] Wobst, Reinhard, a.a.O., Sn. 161, 166 und 199,
  - [#56] Schnoor, „ [www.telecypher.net/CYPHKERN.HTM](http://www.telecypher.net/CYPHKERN.HTM)“
  - [#57] Schneier, Bruce, a.a.O., S. 543,
  - [#58] Schneier, Bruce, a.a.O., S.545, Wobst, Reinhard, a.a.O., S.156  
und Schmeh, Klaus, a.a.O., S.98).
-