

Facharbeit Mathematik - Kryptologie

Thorsten Ferres
MSS94

| | |
|--|-----------|
| <i>Facharbeit Mathematik - Kryptologie</i> | <u>1</u> |
| <i>Kryptologie - Was ist das für eine Wissenschaft?</i> | <u>3</u> |
| <i>Welche Algorithmen gibt es - was sind Algorithmen?</i> | <u>5</u> |
| <i>Substitutionsalgorithmen</i> | <u>6</u> |
| <i>Kryptoanalyse - oder wie ich mir die Freude an der Kryptographie verderben kann</i> | <u>10</u> |
| <i>Der Vigenère-Chiffre als Beispiel für einen polyalphabetischen Chiffrieralgorithmus</i> | <u>12</u> |
| <i>Kryptoanalyse Teil II</i> | <u>15</u> |
| <i>Das Ende des Vigenère-Chiffre</i> | <u>15</u> |
| <i>Die 'Herstellung' einer scheinbar zufälligen Buchstabenfolge als Schlüssel für einen Vigenère-Chiffre</i> | <u>20</u> |
| <i>Doppelt verschlossen - asymmetrische Kryptosysteme</i> | <u>24</u> |
| <i>Schlußbemerkung</i> | <u>32</u> |
| <i>Quelle</i> | <u>33</u> |

Kryptologie - Was ist das für eine Wissenschaft?

Die Kryptologie ist, wie ich schon in der Überschrift verraten habe, eine Wissenschaft. Doch um welche Wissenschaft es sich dabei handelt, ist wohl den meisten unbekannt, allein der Name klingt schon geheimnisvoll genug. Doch mit 'geheimnisvoll' liege ich schon gar nicht so schlecht - die Kryptologie ist nämlich die Wissenschaft des Verheimlichens. Aber was soll hier verheimlicht werden?

Es gibt immer und überall Nachrichten und Informationen, die nicht an dritte weitergegeben werden dürfen, die nur für Sender und Empfänger verständlich sein müssen. Und hier setzt die Kryptologie ein, genauer gesagt die Kryptographie, so heißt meist der Teil dieser Wissenschaft, der sich mit dem Verschlüsseln von Botschaften befaßt. Der andere Teil, die Kryptoanalyse, beschäftigt sich genau mit dem Gegenteil, nämlich dem Entschlüsseln von Nachrichten.

Soll nun eine Botschaft verschlüsselt werden, so müssen Sender und Empfänger zwei ganz entscheidende Dinge wissen:

1. Sie müssen sich darüber einig sein, auf welche Art die Botschaft verschlüsselt werden soll, d. h. beide müssen den Algorithmus kennen, mit dem verschlüsselt wird.
2. Außerdem sollten Sie ein Schlüssel vereinbaren. Dieser Schlüssel kann nun in den Algorithmus eingesetzt werden, um den Text zu 'verschließen'. Der Empfänger besitzt natürlich auch einen Schlüssel, er muß den Text ja wieder 'aufschließen' können.

Doch warum Algorithmus und Schlüssel - würde ein Algorithmus mit einem festen Schlüssel nicht ausreichen?

Ziel des Verschlüsseln ist es, Botschaften geheim zu halten. Wenn ich nun einen Algorithmus mit festem Schlüssel verwende, so kann jeder, der diesen Algorithmus kennt, die Nachricht wieder entschlüsseln.

Wenn wir aber einen Algorithmus verwenden, für den es viele verschiedene Schlüssel gibt, so muß jeder, der sich den Text aufschlüsseln möchte, auch diesen ganz bestimmten Schlüssel kennen, mit dem der Text verschlüsselt wurde. Die Kenntnis allein über den Algorithmus hilft ihm hier nicht weiter.

Es handelt sich hierbei also um das Problem der Sicherheit. Allgemein läßt sich darüber folgendes aussagen: Die Sicherheit eines System darf nicht allein von der

Geheimhaltung des Algorithmus abhängen, sie sollte nur von der Geheimhaltung des verwendeten Schlüssels abhängig sein.

Welche Algorithmen gibt es - was sind Algorithmen?

Was ist ein Algorithmus - dieser Begriff soll hier zunächst einmal geklärt werden. Nehmen wir einmal ein Beispiel, das wohl jedem von uns bekannt sein sollte. Die Mutter macht Pudding. Sie geht dabei in einer ganz bestimmten Reihenfolge vor. Zunächst gibt sie Milch in eine Schüssel, dann gibt sie das Puddingpulver hinzu, vielleicht noch etwas Zucker und schließlich rührt sie das Gemisch solange, bis der Pudding steif ist. So geht sie jedesmal vor, wenn sie Pudding macht, egal ob Schokoladen - oder Vanillepudding, sie nimmt nur bei der Zubereitung des entsprechenden Puddings das entsprechende Pulver. Eine solche Tätigkeit, bei der immer wieder dieselben Arbeitsschritte in derselben Reihenfolge ausgeführt werden, wird Algorithmus genannt.

Nun, welche Algorithmen kommen beim Verschlüsseln zum Zuge. Zunächst möchte ich einmal zwei ganz grundlegende Arten der Verschlüsselung unterscheiden:
























1. Es gibt sogenannte Transpositionsalgorithmen, bei denen jeder Buchstaben seine Bedeutung behält, nur seine Position innerhalb des Textes so verändert, daß der Text nicht mehr lesbar ist.
2. Des weiteren sind sogenannte Substitutionsalgorithmen bekannt, mit denen verschlüsselt werden kann. Hierbei behält der Buchstabe zwar seine Position bei, wird aber durch einen bestimmten Geheimbuchstaben ersetzt, so daß ein Unbefugter wiederum die Nachricht nicht entziffern kann.

Die Transpositionsalgorithmen möchte ich hier einmal zurückstellen. Wer sich trotzdem dafür interessiert, den möchte ich darauf aufmerksam machen, daß er/sie sie ja eigentlich schon kennt, wenn er nur einmal versucht hat, aus einem Buchstabensalat einen sinnvollen Text zu machen. Das Durcheinander der Buchstaben kann durch einen solchen Transpositionsalgorithmus erreicht werden. (Doch sind in solchen Rätseln, um sie noch zu erschweren, die Positionen der einzelnen Buchstaben meist per Zufall festgelegt worden, so daß ich kaum noch von einem Algorithmus sprechen könnte.)

Substitutionsalgorithmen

Im weiteren Verlauf dieser Facharbeit möchte ich Euch/Ihnen nun die Möglichkeiten der Substitutionsalgorithmen näherbringen.

Es gibt verschiedene Möglichkeiten, um mittels einen solchen Algorithmus zu verschlüsseln. Einmal könnte ich beispielsweise hingehen und jedem Buchstaben des Alphabetes ein ganz bestimmtes Geheimzeichen zuordnen, wie ich es in der folgenden Tabelle getan habe:

| Buchstabe | Geheimzeichen |
|-----------|---|
| 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 | ☾ |

Aus dem folgenden Klartext, so heißt im Allgemeinen den Text, den ich verschlüsseln möchte,

KRYPTOLOGIE IST EINE SCHOENE WISSENSCHAFT

erhalte ich mit der in der Tabelle oben aufgezeigten Kodierung folgenden Geheimtext:



Anmerkung: Die Umlaute ‘Ä’, ‘Ö’ und ‘Ü’ werden meist wie ‘AE’, ‘OE’ und ‘UE’ behandelt.

Wie jeder unschwer erkennen kann, ist dieser Text nun nicht mehr lesbar und kann von niemandem, der die Tabelle oben nicht kennt, wieder entschlüsselt werden.

Doch genau hier liegt das Problem dieser Verschlüsselungsmethode. Der Empfänger einer solchen Nachricht muß eben auch die gesamte Tabelle kennen, um die für ihn bestimmte Botschaft lesen zu können. Und wenn wir uns die einzelnen Geheimzeichen genauer anschauen, so wird schnell auffallen, daß einige der Zeichen sich sehr ähnlich sehen, so daß bei einer Verschlüsselung per Hand sehr genau darauf geachtet werden müßte, daß das Geheimzeichen für ‘U’ nicht genauso aussieht wie das Geheimzeichen für ‘V’ (vgl. die beiden Geheimzeichen in der Tabelle oben). Jetzt kann ich auch das Geheimnis lüften, wie ich zu dieser Tabelle gekommen bin. Meinem Textverarbeitungsprogramm stehen sehr viele verschiedene Schriftarten zur Verfügung. Darunter sind allerdings auch einige, die nicht aus lauter Buchstaben, sondern aus lauter Zeichen bestehen. Eine solche Schriftart habe ich zum Verschlüsseln benutzt. Wenn ich dem Empfänger den Geheimtext in digitaler Form z. B. als Textdatei dieses Textverarbeitungsprogrammes zukommen lassen, müßte er die obige Tabelle nicht einmal kennen, sondern könnte den Geheimtext einfach in eine andere Schriftart umwandeln - und im Handumdrehen hätte er den lesbaren Klartext vor sich. Doch wenn dritte diesen Geheimtext abfangen, wäre es auch ihnen ein Leichtes, den Text auf diese Weise zu entziffern.

Wir müssen also eine Möglichkeit finden, den Text so zu verschlüsseln, daß der Empfänger ihn mittels einer weiteren, kurzen Information, die dann allerdings sehr geheim gehalten werden muß, entziffern kann. Diese weitere Information, die der Empfänger erhält, ist der sogenannte Schlüssel.

Als erstes weisen wir den einzelnen Buchstaben des Alphabetes jeweils eine Zahl zu: A=1, B=2, C=3,..., X=24, Y=25, Z=0. Nun bestimmen wir einen Buchstaben zu unserem Schlüssel. Die Zahl dieses Buchstabens addieren wir zu den Nummern der einzelnen Buchstaben des Klartextes und nehmen dann den Buchstaben als Geheimbuchstaben, der die erhaltene Nummer hat. Sollten sich dabei Zahlen ergeben, die größer als 25 sind (Nummer des Buchstabens 'Y', größte vorkommende Zahl), so muß ich den Rest nehmen, den ich bei der Division dieser Zahl durch 26 erhalte. (Warum durch 26? - Weil das Alphabet 26 Buchstaben hat und ich durch diese Rechnung keine Zahl erhalten kann, die größer als 25 ist.) Den Rest einer Division erhalte ich durch die sogenannte Modulfunktion. Ich schreibe also wie folgt:

$$\text{Rest} = a \text{ mod } b$$

Wenn ich einen Schlüsselbuchstaben festgelegt habe, kann ich mir so ebenfalls eine Tabelle anlegen, die angibt, welcher Klartextbuchstabe welchem Geheimtextbuchstaben entspricht.

Verschlüsselung mit dem Schlüsselbuchstaben 'W':

| | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Klralphabet | 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 | 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 |

Auch hierfür möchte ich ein Verschlüsselungsbeispiel angeben:

Klartext:

VERSCHLUESSELN IST NICHT SCHWER

Geheimtext:

sbopzeirbppbik fpq kfzeq pzetbo

Jetzt wurde die Buchstaben zwar wiederum in Buchstaben verwandelt, doch in dieser Reihenfolge kann wohl niemand so schnell etwas damit anfangen. Weil bei dieser Chiffriermethode jedem Buchstaben ein Buchstabe zugewiesen wird, der um die durch den Schlüssel bestimmte Anzahl von Stellen vor oder nach diesem Buchstaben steht, also das Alphabet irgendwie verschoben wurde, heißen diese Algorithmen Verschiebechiffren.

Aber behalten die Buchstaben auch im Geheimalphabet die gleiche Reihenfolge bei, so folgt auf 'A' der Buchstabe 'B' usw. Natürlich kann ich bei meinem Geheimalphabet auch die Buchstaben in anderer Reihenfolge anordnen. Dies

erreiche ich, indem ich die dem Buchstaben zugewiesene Zahl jeweils mit einer ganz bestimmten Zahl multipliziere und den Rest dieser Zahl bei der Division durch 26 nehme, um zu dem Geheimtextbuchstaben zu kommen.

Allerdings muß ich hier darauf achten, daß die Chiffrierung eindeutig ist, multipliziere ich z. B. mit der Zahl 2, so wird schnell auffallen, daß sowohl dem Buchstaben 'E' als auch dem Buchstaben 'R' der Geheimbuchstabe 'j' zugewiesen werden. Eine solche Chiffrierung ließe sich nicht mehr eindeutig entschlüsseln, woher soll ich wissen, ob diese 'j' nun von einem 'E' oder von einem 'R' herrührt. Die Zahlen mit denen wir multiplizieren können, ohne daß solche Doppelnennungen vorkommen, sind **1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25**. Das sich ausgerechnet diese Zahlen als Faktor verwenden lassen, hängt damit zusammen, daß in keiner dieser Zahlen Teiler der Zahl 26 vorkommen. Verknüpfen wir die beiden Methoden (die 26 Verschiebealgorithmen und die 12 Multiplikationsmethoden) so erhalten wir

$$12 \cdot 26 = 312$$

verschiedene Verschlüsselungsmöglichkeiten. Angeben kann ich die Verschlüsselung ganz einfach durch zwei Zahlen, z. B. [3, 7], wobei die erste die Verschiebung angibt, die zweite den Faktor, mit dem anschließend multipliziert wird.

Doch auch wenn diese Menge von 312 Verschlüsselungen schon relativ groß scheint, sind das nicht sonderlich viele Möglichkeiten, denn die heutigen Computer haben sie sehr schnell alle durchprobiert. Allerdings müßte bei jedem Versuch ein Mensch prüfen, ob es sich bei dem entschlüsselten Text wirklich um den Klartext handelt, ein Computer kann ja nicht lesen. Doch lassen sich solche Chiffrierungen auch auf andere Weise sehr leicht knacken.

Kryptoanalyse - oder wie ich mir die Freude an der Kryptographie verderben kann

Nachdem wir uns lange genug damit beschäftigt haben, unsere Texte zu verschlüsseln, kommen wir nun zur zweiten Teilwissenschaft der Kryptologie - der Kryptoanalyse. Wie gehe ich bei der Analyse eines Geheimtextes vor, wenn ich den Schlüssel nicht kenne?

Dazu müssen wir uns die Sprachen, in erster Linie meine ich hier natürlich die deutsche Sprache, genauer ansehen. Wenn wir uns diesen oder auch jeden anderen Text anschauen, der in deutscher Sprache verfaßt ist, so ist es ganz offensichtlich, daß der häufigste Buchstabe ohne Zweifel das 'E' ist. Durch allgemeine Statistiken läßt sich die Häufigkeit aller Buchstaben herausfinden, so daß ich sie in einer Tabelle zusammenstellen konnte, die nun (für die deutsche Sprache) allgemeine Gültigkeit hat, auch wenn es natürlich Texte oder Textabschnitte geben mag, die eine ganz andere Häufigkeitsverteilung aufweisen. (Bsp.: Zehn zahme Ziegen zogen zehn Zentner Zucker zum Zuericher Zug. / Xaver spielt gut Xylophon.)

| Buchstabe | Häufigkeit (in %) | Buchstabe | Häufigkeit (in %) |
|-----------|-------------------|-----------|-------------------|
| a | 6,51 | n | 9,78 |
| b | 1,89 | o | 2,51 |
| c | 3,06 | p | 0,79 |
| d | 5,08 | q | 0,02 |
| e | 17,40 | r | 7,00 |
| f | 1,66 | s | 7,27 |
| g | 3,01 | t | 6,15 |
| h | 4,76 | u | 4,35 |
| i | 7,55 | v | 0,67 |
| j | 0,27 | w | 1,89 |
| k | 1,21 | x | 0,03 |
| l | 3,44 | y | 0,04 |
| m | 2,53 | z | 1,13 |

Mit Hilfe dieser Tabelle ist es mir nun möglich, für einen bestimmten Geheimtextbuchstaben den entsprechenden Klartextbuchstaben zu finden. Ich muß nur seine Häufigkeit im gesamten Text bestimmen und dann in dieser Tabelle nachschauen, welche Buchstabe eine solche Häufigkeit aufweist. Dazu gehe ich wie folgt vor: Ich zähle die Anzahl der Buchstaben in meinem Geheimtext, deren Häufigkeit ich bestimmen möchte. Die nun erhalte Zahl teile ich durch die Anzahl

der Buchstaben des gesamten Textes und erhalte somit die Häufigkeit dieses Buchstabens.

Wenn ich sogar noch sicher sein kann, daß es sich bei der verwendeten Verschlüsselungsmethode um einen einfachen Verschiebechiffre handelt, das soll heißen, der Sender der Nachricht hat die einzelnen Buchstaben nur verschoben, aber nicht mit einem Faktor multipliziert, so wird es meist ausreichen, das Geheimäquivalent des Buchstabens 'E' zu finden, um festzustellen, wie auch alle anderen Buchstaben verschlüsselt wurden.

Wie wir gesehen haben, ist eine solche Verschlüsselungsmethode nicht als sicher zu bezeichnen. Doch warum ist dies so? Alle gleichen Klartextbuchstaben werden immer durch denselben Geheimtextbuchstaben verschlüsselt, so daß die Häufigkeitsverteilung der Buchstaben gleich bleibt. Solche Chiffriermethoden werden auch gerne als monoalphabetisch bezeichnet.

Der Vigenère-Chiffre als Beispiel für einen polyalphabetischen Chiffrieralgorithmus

Eine andere Verschlüsselungsmethode, die ich im folgenden vorstellen möchte, geht so vor, daß nicht jeder Buchstabe um die gleiche Anzahl verschoben wird, sondern der jeweils folgende Buchstabe wird immer um eine andere Anzahl verschoben. Wie kann ich so etwas erreichen? Schon im Jahre 1586 kam der französische Diplomat Blaise de Vigenère auf eine Lösung, mit der sich dieses Problem beheben läßt. Es wird ganz einfach eine bestimmte Reihenfolge festgelegt, mit der die einzelnen Buchstaben verschoben werden.

Wie bereits bei den Verschiebealgorithmen gesagt wurde, läßt sich auch ein Buchstabe festlegen, um mit dessen Nummer zu verschieben. Hier wird zum Verschieben nicht ein einzelner Buchstabe verwendet, sondern gleich mehrere, ein ganzes Wort. Dazu einigen sich Sender und Empfänger auf ein Schlüsselwort, zum Verschlüsseln schreibe ich dieses Wort solange über den Klartext, bis dessen Ende erreicht ist. Nun habe ich über jedem Buchstaben des Klartextes den Schlüsselbuchstaben stehen, mit dem verschoben werden muß, um zum Geheimtext zu gelangen. Beim Entschlüsseln gehe ich natürlich entsprechend vor.

Vigenère hatte sogar den Einfall - es handelt sich bei den einzelnen Verschlüsselungen ja eigentlich nur um reine Verschiebechiffren, von denen es bekanntlich ja nur 26 gibt - alle möglichen Geheimalphabete in einer Tabelle zusammenzufassen, um diese Verschlüsselungsmethode noch zu vereinfachen. Seine Tabelle sieht nämlich so aus, daß ich beim Verschlüsseln denjenigen Buchstaben als Geheimbuchstaben nehme, der in der Zeile, die mit dem Schlüsselbuchstaben beginnt, unter dem zu verschlüsselnden Buchstaben steht. Zum besseren Verständnis möchte ich hier nun das nach seinem Erfinder Vigenère benannte Quadrat abdrucken:

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 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 | w | 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 |

Beim Verschlüsseln gehe ich also so vor, daß ich zuerst in der ersten Zeile nach dem Buchstaben suche, den ich verschlüsseln möchte. Nun suche ich in der ersten Spalte nach dem Buchstaben, der als Schlüssel darüber steht. In dieser Zeile suche ich jetzt nach dem Buchstaben, der in derselben Spalte wie mein zu verschlüsselnder Buchstabe steht und erhalte somit den Geheimbuchstaben.

Diese Beschreibung klingt etwas seltsam, doch darum möchte ich auch hier wiederum ein Beispiel angeben, aus dem dann aber sehr leicht ersichtlich werden sollte, wie ich bei dieser Chiffrierung vorgehen muß.

Beispiel:

Klartext:

DIES IST EIN POLYALPHABETISCHER CHIFFRE

Schlüssel:

ROSA

Verschlüsselung:

Schlüssel: rosa ros aro sarosarosarosarosa rosaros
 Klartext: DIES IST EIN POLYALPHABETISCHER CHIFFRE
 Geheimtext: **uwws zgl ezb hocmslgvsbvhastvwr tvafwfw**

Hier fällt sicherlich gleich auf, daß beispielsweise der Geheimbuchstabe ‘w’ einmal durch eine Verschlüsselung des Buchstabens ‘I’, doch gleich dahinter durch die Verschlüsselung des Buchstabens ‘E’ erreicht wurde. So erreiche ich, daß die Häufigkeitsverteilung der Buchstaben, wie sie in einem normalen deutschen Text vorkommt, nicht mehr erhalten bleibt. Deshalb kann ich sagen, daß diese Chiffrieremethode sicherer als eine monoalphabetische ist. Doch ist sie

deshalb noch nicht knackbar? - Doch, auch den Vigenère-Algorithmus kann ich allein durch Kryptoanalyse entschlüsseln, ohne den Schlüssel zu kennen.

Kryptoanalyse Teil II

Das Ende des Vigenère-Chiffre

Bei der Analyse eines mittels des Vigenère-Chiffre verschlüsselten Textes gehen wir ebenfalls mit Hilfe der statistischen Häufigkeit der Buchstaben vor. Momentmal, habe ich nicht oben erst gesagt, daß die Häufigkeitsverteilung der Buchstaben in einem mit dieser Methode verschlüsselten Text nicht mehr stimmt? Das ist richtig, doch nur wenn ich mir den gesamten Text anschau.

Aber es werden doch immer wieder Buchstaben mit dem gleichen Schlüssel verschlüsselt. Welche Buchstaben sind das? Nun, hier spielt die Länge des Schlüsselwortes eine entscheidende Rolle. Denn alle Buchstaben, die unter dem ersten, zweiten, dritten usw. Buchstaben des Schlüsselwortes stehen, werden immer wieder um die gleiche Anzahl verschoben.

Würde ich also die Länge des Schlüsselwortes kennen, könnte ich die Buchstaben, die mit demselben Schlüssel verschlüsselt wurden, in eine Reihe schreiben, und in den einzelnen Reihen gilt dann die Häufigkeitsverteilung der Buchstaben wie in der Sprache, in der der Klartext geschrieben wurde. Für die deutsche Sprache heißt dies, daß es sogar ausreichen würde, das Geheimäquivalent des Buchstabens 'E' zu finden, um mit diesem Wissen auf den Schlüssel zu schließen, womit auch die anderen Buchstaben entschlüsselt werden könnten.

Das bedeutet für unser Beispiel von oben:

```

R   u z z c g v t t w
O   w g b m v h v v f
S   w l h s s a w a w
A   s e o l b s r f

```

Doch wie kann ich allein durch Kenntnis eines Geheimtextes auf die Länge des Schlüsselwortes schließen? Dazu gibt es zwei Möglichkeiten, die beide nach ihren Erfindern benannt wurden: Den Kasiski-Test und den Friedmann-Test.

Hier zunächst die etwas einfachere Testmethode.

Der Kasiski-Test:

Kommen in einem Geheimtext häufig gleiche Buchstabenfolgen vor, so kann ich hoffen, daß diese dadurch zustande kamen, weil es sich erstens um die gleichen Wörter handelt (z. B. Wörter wie 'der, die das, ein, einer, eine' kommen in einem Text sehr häufig vor) und zweitens weil sie mit denselben Schlüsseln verschlüsselt

wurden. Doch das ist nur der Fall, wenn der Schlüssel genau n -mal zwischen diese Folgen paßt (n sei irgendeine natürliche Zahl). Zählen wir nun den Abstand zweier solcher Buchstabenfolgen aus (Achtung: eine der Buchstabenfolgen muß ganz mitgezählt werden, die andere hingegen darf nicht mitgezählt werden!), so erhalten wir die Länge des Schlüsselwortes, indem wir durch n teilen. Wenn wir diesen Test für mehrere Buchstabenfolgen durchführen, so wird das n zwar jedes Mal verschieden sein, doch müssen die Abstände immer noch durch die Länge des Schlüsselwortes teilbar sein. Am besten zerlege ich die Abstände in Primfaktoren, so daß ich sehr leicht sehen kann, welche Faktoren in den meisten Fällen vorkommen, d. h. welche Schlüsselwortlänge am wahrscheinlichsten ist. Ich sage wahrscheinlich, denn die Länge des Schlüsselwortes kann auch Teiler enthalten, die zwar nicht überall, aber doch recht häufig vorkommen. Die anderen Buchstabenfolgen könnten ja auch zufällig zustande gekommen sein. Dieser Test liefert also nur die Schlüsselwortlänge auf Teiler, die in ihr enthalten sind.

Nun die etwas schwierigere Methode.

Der Friedmann-Test:

Diese Methode ist deshalb etwas schwieriger, weil sie sehr viel Wahrscheinlichkeitsberechnung enthält. Ihr Vorteil besteht jedoch darin, daß sie uns eine Formel liefert, mit der wir ohne großartigen Aufwand die Länge des Schlüsselwortes bestimmen können.

Befassen wir uns zunächst einmal mit der Wahrscheinlichkeit, daß zwei Buchstaben gleich sind. Gegeben sei ein Text der Länge n Buchstaben. Für die Wahl des ersten Buchstabens habe ich n Möglichkeiten. Für die Wahl des zweiten Buchstabens bestehen nur noch $n-1$ Möglichkeiten. Ich habe also $\frac{n \cdot (n-1)}{2}$ Möglichkeiten, um zwei Buchstaben aus diesem Text zu wählen. (Ich muß noch durch zwei teilen, da es ja zwei Möglichkeiten gibt, um zwei Buchstaben zu wählen: zuerst den ersten, dann den zweiten oder zuerst den zweiten und dann den ersten - obwohl es sich in beiden Fällen um die gleichen Buchstaben handelt.) Die Wahrscheinlichkeit, daß der erste gewählte Buchstabe ein 'A' ist, beträgt p_1 . Ebenso ist die Wahrscheinlichkeit, daß auch der zweite Buchstabe ein 'A' ist, ungefähr p_1 . (Sie ist genau p_1 , wenn ich auch die gleiche Stelle wählen darf. Doch bei sehr großen Texten ist der dadurch entstehende Fehler so klein, daß ich ihn vernachlässige.) Also ist die Wahrscheinlichkeit, daß es sich in beiden Fällen um den Buchstaben 'A' handelt, p_1^2 . Das Entsprechende gilt auch für alle anderen Buchstaben. Wenn es mir nur wichtig ist, daß diese beiden Buchstaben gleich sein sollen, so muß ich die Summe der einzelnen Wahrscheinlichkeiten p_1^2, p_2^2, p_3^2 bis p_{26}^2 bilden, und erhalte so die Wahrscheinlichkeit, daß es sich bei beiden Buchstaben um die gleichen handelt:

$$p_1^2 + p_2^2 + p_3^2 + p_4^2 + p_5^2 + \dots + p_{25}^2 + p_{26}^2 = \sum_{i=1} p_i^2$$

Für einen in deutscher Sprache verfaßten Text heißt das, ich muß die Quadrate der in der Tabelle oben aufgeführten Häufigkeitswerte addieren und erhalte:

$$\sum_{i=1}^{26} p_i^2 = 0,0762$$

Diese Zahl sagt jetzt folgendes aus: Wenn ich einen deutschen Text vor mir habe und wähle per Zufall zwei Buchstaben daraus aus, so kann ich mit 7,62%iger Wahrscheinlichkeit sagen, daß diese beiden Buchstaben gleich sind.

Aber diese Aussage gilt nur für ganz normale, deutsche Texte. Doch nehmen wir an, wir haben ein Buchstabensalat vor uns, in dem jeder Buchstabe mit der gleichen Wahrscheinlichkeit vorkommt:

$$p_i = \frac{1}{26}$$

Die Wahrscheinlichkeit, in einem solchen Buchstabensalat zwei gleiche Buchstaben zu wählen, ist:

$$\sum_{i=1}^{26} p_i^2 = \sum_{i=1}^{26} \frac{1}{26^2} = 26 \cdot \frac{1}{26^2} = \frac{1}{26} \approx 0,0385$$

Die beiden gefunden Zahlen werden als sogenannte (Friedmannsche) Koinzidenzindexe bezeichnet. Diese Überlegungen werden einigen wohl etwas seltsam vorkommen, aber wir werden sie jetzt gleich benötigen.

Zurück zu den Chiffrierungen und zur Kryptoanalyse. Nehmen wir an, ich wähle mir aus einem mittels Vigenère-Chiffre verschlüsselten Geheimtext der Länge n einen Buchstaben. Nun suche ich mir aus den übrigen Buchstaben noch einen zweiten aus. Dazu habe ich wie bereits weiter oben erwähnt $\frac{n \cdot (n-1)}{2}$ Möglichkeiten.

Doch mit welcher Wahrscheinlichkeit kann ich sagen, daß diese beiden Buchstaben gleich sind. Und hier kommt wieder die Länge des Schlüsselwortes ins Spiel. Ich habe ebenfalls schon gesagt, daß ich die Buchstaben in Reihen

schreiben kann, so daß alle Buchstaben in einer Reihe mit dem gleichen Schlüssel verschlüsselt wurden. Die Anzahl dieser Reihen spiegelt die Länge des Schlüsselwortes wieder. Innerhalb der Reihe gilt ja die Häufigkeitsverteilung der Buchstaben in der (hier: deutschen) Sprache. Doch innerhalb einer Reihe habe ich auch nur $\frac{n \cdot (n/l - 1)}{2}$ Möglichkeiten, um zwei Buchstaben zu wählen (l sei die Länge des Schlüsselwortes, n die Länge des gesamten Geheimitextes - also erhalte ich die Anzahl der Buchstaben, die in einer Reihe stehen, indem ich n durch l teile). Die Wahrscheinlichkeit, daß nun zwei Buchstaben in einer Reihe gleich sind, ist

$$\frac{n \cdot (n - l)}{2l} \cdot 0,0762,$$

denn hier ist ja die Häufigkeit der einzelnen Buchstaben wie in der deutschen Sprache (daher der Faktor 0,0762).

Es besteht aber auch noch die Möglichkeit, außerhalb der gleichen Reihe meinen zweiten Buchstaben zu wählen. Die Zahl der Möglichkeiten ist hier

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

Auch hier ist wiederum die Wahrscheinlichkeit interessant, daß zwei Buchstaben gleich sind, doch hier ist sie viel geringer, denn die Häufigkeit der Buchstaben wird ja durch die verschiedenen Schlüssel verändert, deshalb muß hier mit dem Faktor 0,0385 multipliziert werden:

$$\frac{n^2 \cdot (l - 1)}{2l} \cdot 0,0385$$

Diese beiden Möglichkeiten müssen nun addiert und durch die gesamten Möglichkeiten, zwei Buchstaben zu wählen, geteilt werden (d. h. Division durch $\frac{n \cdot (n - 1)}{2}$), der Index wird also nach dem Motto Anzahl der günstigen Fälle durch Anzahl der möglichen Fälle berechnet.

Den Koinzidenzindex bezeichnen wir nun im folgenden mit dem Buchstaben **I**:

$$I = \frac{\frac{n \cdot (n - l)}{2l} \cdot 0,0762 + \frac{n^2 \cdot (l - 1)}{2l} \cdot 0,0385}{\frac{n \cdot (n - 1)}{2}} = \frac{(n - l) \cdot 0,0762 + n \cdot (l - 1) \cdot 0,0385}{l \cdot (n - 1)} = \frac{1}{l \cdot (n - 1)} \cdot [0,0377n + l(0,0385n - 0,0762)]$$

Wir lösen diese Formel nach der Länge l des Schlüsselwortes auf und erhalten eine neue Formel, in die wir nur noch die Länge des Geheimtextes und die Häufigkeit, daß zwei Buchstaben gleich sind (durch Abzählen und entsprechendes Verrechnen, d. h. Addition der Quadrate der Häufigkeit der im Geheimtext auftauchenden Buchstaben), einsetzen müssen:

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

Das ‘ungefähr’-Zeichen steht deshalb, weil wir in unserer Formel mit allgemeinen Häufigkeitskonstanten arbeiten, die aber nicht für jeden Text gültig sein können. Trotzdem haben wir mit dieser Formel ein starkes Werkzeug in der Hand, um jeden Vigenère-Chiffre zu knacken.

Wirklich jeden? Nun, es lassen sich leider nicht alle auf diese Weise entschlüsseln, nur solche, die mittels eines kurzen Schlüsselwortes verschlüsselt wurden. Daraus folgt also, daß Chiffren mit sehr langem Schlüsselwort sicherer sein müssen. Aber wie lang muß mein Schlüsselwort sein, damit die Chiffrierung sicher ist. Ganz einfach - am besten ist der Schlüssel so lange wie der Klartext.

Doch wie soll ich einen solchen Schlüssel übermitteln? - Beispielsweise könnte ich dem Empfänger Titel, Autor und Seiten eines Buches mitteilen, den er als Schlüssel verwenden soll. Er bekommt also nur eine kurze, sicher übermittelbare Information. Doch hierbei ist der Schlüssel ein deutscher Text, der die statistischen Häufigkeiten aufweist, womit ein Kryptoanalytiker bei seiner Analyse schon etwas anfangen könnte.

Die andere, nun wirklich sichere Lösung für einen Schlüssel wäre eine zufällige Folge aus Buchstaben, wie ich sie z. B. durch einen Zufallsgenerator, der ganz willkürlich einen Buchstaben ausspuckt, erhalten würde. Eine solche, völlig zufällig zustande gekommene Folge aus Buchstaben wird Buchstabenwurm genannt. Doch hier besteht wieder das Problem der Schlüsselübermittlung, auch wenn ein mittels Buchstabenwurm und Vigenère-Chiffre verschlüsselter Text perfekte Sicherheit böte, denn ein Analytiker kann nie sagen, daß ein bestimmter Geheimtextbuchstabe durch einen bestimmten Schlüssel codiert wurde, da alle Schlüssel mit der gleichen Wahrscheinlichkeit auftauchen. Er kann z. B. nur sagen, daß ein bestimmter Geheimbuchstabe mit 17,4%iger Wahrscheinlichkeit von einem ‘E’ herrührt, doch das gilt ja auch für alle anderen Buchstaben des übrigen Geheimtextes und hilft ihm somit nicht weiter.

Die 'Herstellung' einer scheinbar zufälligen Buchstabenfolge als Schlüssel für einen Vigenère-Chiffre

Über den Vigenère-Chiffre haben wir schon ganz am Anfang gesagt, daß er nicht mehr die Buchstabenhäufigkeit der deutschen Sprache widerspiegelt. Also würde ein solcher Vigenère-Chiffre, wenn ich ihn als Schlüssel verwenden würde, auch kaum mehr statistisch erfaßbare Daten enthalten.

Das Problem liegt bei der Sache so, daß ich ja als Schlüssel einen möglichst langen Text benötige, um eine größtmögliche Sicherheit zu erhalten, obwohl der Schlüssel auf der anderen Seite relativ kurz sein muß, um sicher übermittelt werden zu können.

Wie wäre es denn, wenn ich aber nun zwei Schlüssel verwenden würde? Einen von diesen benötige ich, um den anderen zu verschlüsseln, den erhaltenen Geheimtext benütze ich als endgültigen Schlüssel.

Aber warum soll mein endgültiger Schlüssel jetzt so lang wie der Geheimtext sein. Beim Verschlüsseln des zweiten Schlüssel schreibe ich nicht nur den ersten Schlüssel solange über den zweiten, bis dieser genauso lang ist, sondern ich schreibe beide Schlüssel solange untereinander, bis beide Reihen die gleiche Länge haben, aber in keiner nur ein Bruchstück eines Schlüsselwortes steht, sondern in beiden die Schlüsselwörter - auch beim letzten Mal - komplett ausgeschrieben wurden.

Deshalb muß ich bei der Wahl meiner Schlüsselwörter darauf achten, daß der kleinste gemeinsame Vielfache der beiden Buchstabenzahlen möglichst so groß oder größer als der zu verschlüsselnde Klartext ist. So erreiche ich nämlich, daß der endgültige Schlüssel so lang bzw. sogar länger als der Klartext ist, ohne daß in ihm selbst wieder irgendwelche Regelmäßigkeiten auftauchen.

Doch warum ist das so? Nehmen wir einmal an, beide Schlüssel bestünden aus verschiedenen Buchstaben, das soll heißen in keinem käme ein Buchstaben doppelt vor. Daraus folgt, daß ein und derselbe Buchstabe des einen Schlüsselwortes immer wieder mit einem anderen Buchstaben des anderen Schlüsselwortes verschlüsselt würde, ein Kryptoanalytiker könnte also nie sagen, daß zwei gleiche Buchstabenfolgen dadurch zustande kamen, weil sie mit dem gleichen Schlüssel verschlüsselt wurden.

Beispiel:

| | |
|---------------------|-----------------------|
| Schlüssel 1: | ELFELFELFELFELFELFELF |
| Schlüssel 2: | HUNDERTHUNDERTHUNDERT |

Endgültiger Schlüssel: LFSHPWXSZROJVEYYIICY

Dieser Schlüssel besteht nun (wenn ich von den beiden 'Y's und 'I's am Ende absehe) aus einer ziemlich wirren, scheinbar zufällig zustande gekommenen Buchstabenfolge, die kaum statistisch erfassbare Daten enthält. Doch kann ich mit ihm auch nur bis zu 21 Buchstaben lange Klartexte verschlüsseln, wenn mein Chiffre sicher sein soll.

Als längste Schlüssel kämen zwei Buchstabenfolgen in Frage, von denen die eine aus 26 verschiedenen Buchstaben bestehen kann, denn es gibt ja bekanntlich nur 26 verschiedene Buchstaben, die andere jedoch nur aus 25, denn die Zahlen 25 und 26 sind teilerfremd ($26=2\cdot 13; 25=5\cdot 5$), so daß ihr kleinster gemeinsamer Vielfache die Zahl $25\cdot 26=650$ ist. Ich könnte also mit einem solchen Schlüssel Klartexte mit bis zu 650 Buchstaben verschlüsseln.

Das Problem ist nur, daß es wohl kaum Wörter geben wird, die aus 26 bzw. 25 verschiedenen Buchstaben bestehen. Doch auch hierfür kann ich eine ziemlich einfache Lösung anbieten. Für ein "Wort" mit 26 verschiedenen Buchstaben muß ich wohl das ganze Alphabet verwenden und die Buchstaben des Alphabets kann ich ja durch Verschieben und Multiplizieren mit einer der 12 Faktoren in 312 verschiedenen Reihenfolgen anordnen. Für die zweite Buchstabenfolge gehe ich genauso vor, nur lasse ich einen beliebigen Buchstaben einfach weg, damit die Buchstabenfolge nur 25 Buchstaben lang wird. Die Anzahl der Schlüssel steigt somit auf

$$26\cdot 12\cdot 25^* \cdot 12\cdot 26=2.433.600$$

* Hier gibt es eigentlich nur noch 25 Verschiebungsmöglichkeiten, da ja am Ende nur mehr 25 Buchstaben übrig bleiben.

Möglichkeiten. Das sind wohl doch eine ganze Menge, die ein Computer zwar schnell durchprobiert hätte, doch muß ein menschlicher Benutzer jedesmal noch überprüfen, ob es sich bei dem entschlüsselten Text wirklich um den Klartext handelt und das kann bei einer solchen doch schon als groß zu bezeichnenden Menge mühsam werden und kostet auch Zeit.

Aber der Vorteil dieser Methode liegt nun darin, daß ich dem Empfänger meiner geheimen Botschaft nicht den endgültigen, langen und ziemlich verwirrenden Schlüssel mitteilen muß, sondern es reicht aus, wenn ich ihm die beiden anderen, kürzeren Schlüssel übergebe, wobei es sogar egal ist, welchen der beiden er zum Verschlüsseln des anderen benutzt, da ja in jedem Fall die gleichen Buchstaben übereinander stehen, d. h. es werden die gleichen Zahlen addiert.

Aber natürlich ist auch eine Buchstabenfolge aus 25 bzw. 26 Buchstaben ziemlich verwirrend, vor allem, wenn sie in keiner bestimmten Reihenfolge stehen. Doch wie bereits weiter oben erwähnt läßt sich ein solches verschobenes und mit einem

der Faktoren multipliziertes Alphabet nur durch zwei Zahlen angeben, wobei die erste für die Verschiebung, die zweite für den Faktor steht. Bei der zweiten Folge muß ich dann noch den Buchstaben bzw. die Nummer des Buchstabens angeben, der wegfallen soll, denn die Folge soll ja nur aus 25 Buchstaben bestehen. Doch eins muß beim Herstellen der beiden Schlüssel beachtet werden: Ich sollte zuerst das Alphabet verschieben und mit dem Faktor multiplizieren, bevor ich den Buchstaben wegfallen lasse, denn lasse ich zuerst den Buchstaben wegfallen, so stimmen nachher die oben genannten zwölf Faktoren nicht mehr - unser neues Alphabet hätte nämlich nur noch 25 Buchstaben. (Es sei denn, die einzelnen Buchstaben behielten ihre ursprüngliche Nummer bei, dann würde auch bei dem entsprechenden Geheimalphabet nur der weggelassene Buchstaben fehlen).

Eine andere Möglichkeit, die diese Verfahrensweise zuließe, wäre natürlich so ausgelegt, daß ich nicht mehr mit den Faktoren, die für 26 Buchstaben ihre Gültigkeit haben, arbeiten würde, sondern mit Faktoren, die für 25 Buchstaben nutzbar sind, die da wären: 1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 13, 14, 16, 17, 18, 19, 21, 22, 23, 24. (25 und alle diese Zahlen sind nämlich teilerfremd). So erhalte ich für die zweite Buchstabenfolge sogar noch eine paar Möglichkeiten mehr (es sind ja schließlich auch mehr Faktoren, nämlich genau 20 verschiedene, möglich), so daß die Anzahl der Buchstabenfolgen, die aus nur 25 Buchstaben bestünden, auf

$$26 \cdot 25 \cdot 20 = 13.000$$

Möglichkeiten wachsen würde. (Zur Erklärung: Die 26 steht für 26 Möglichkeiten, einen Buchstaben wegfallen zu lassen, die 25 für die nun noch 25 verschiedenen Verschiebungen und die 20 aufgrund der 20 möglichen Faktoren.) Allerdings muß hier, nachdem ein Buchstabe entfallen ist, das neue Alphabet noch einmal durchnummeriert werden.

Diese Zahl muß jetzt noch mit den 312 Möglichkeiten, die ich für die andere Buchstabenfolge habe, multipliziert werden, und ich erhalte

$$13.000 \cdot 312 = 4.056.000$$

Möglichkeiten, wie wir sehen, sind das über eine Million mehr Möglichkeiten als vorhin.

Aber auch diese vielen Möglichkeiten lassen sich wiederum mit nur fünf Zahlen exakt beschreiben. Es genügt also, dem Empfänger fünf Zahlen zu nennen: Zunächst zwei zur Verschiebung und Multiplikation der Buchstabenfolge aus 26 Buchstaben, dann eine, die beschreibt, welchen Buchstaben ich im zweiten Alphabet weglassen möchte und schließlich noch die letzten beiden, die

beschreiben, wie dieses Alphabet aus jetzt nur noch 25 Buchstaben verschoben und mit welchem Faktor es multipliziert wird.

Doppelt verschlossen - asymmetrische Kryptosysteme

Die Verschlüsselungsmöglichkeiten, die wir bisher behandelt haben, waren alle so aufgebaut, daß sowohl Sender als auch Empfänger den gleichen Schlüssel benutzen, um zu chiffrieren bzw. zu dechiffrieren. Diese Kryptosysteme werden als symmetrisch bezeichnet, weil Sender und Empfänger mit dem gleichen Schlüssel arbeiten. Doch gibt es auch Möglichkeiten, daß der Empfänger die Nachricht mit einem anderen Schlüssel wieder entschlüsselt und nicht mit dem gleichen, mit dem der Sender sie verschlüsselt hat. Diese Kryptosysteme werden logischerweise als asymmetrisch bezeichnet.

Doch wie ist es möglich, daß ich mit dem gleichen Schlüssel vom Geheimtext nicht mehr zum Klartext zurückkehren kann? Hierbei bediene ich mich sogenannter Einwegfunktionen. Das sind Funktionen, die sich zwar in eine Richtung gut und einfach durchführen lassen, doch die Ausführung dieser Funktionen in die umgekehrte Richtung ist mit großen Schwierigkeiten verbunden. Ein einfaches Beispiel wäre die Funktion x^2 . Zwei gleiche Zahlen lassen sich ohne Probleme miteinander multiplizieren, doch wenn ich eine Zahl habe, die ich nun so in zwei Faktoren zerlegen soll, daß beide genau gleich groß sind, so ist dies nicht mehr so ohne weiteres machbar. Auch alle anderen Potenzfunktionen weisen die gleichen Schwierigkeiten auf, wenn ich sie wieder umkehren möchte, so daß ich bei Potenzfunktionen von Einwegfunktionen sprechen kann.

Mit einer solchen Einwegfunktion bearbeite ich nun meinen Geheimtext bzw. die Zahl, die meinem Geheimtext zugewiesen wurde. Meist wird nämlich nicht jedem Buchstaben eine eigene Zahl zugewiesen, sondern ich nehme mir eine ganze Buchstabenkolonne, dann schreiben ich die Kodierung der einzelnen Buchstaben wie sie der ASCII-Code vorschreibt in binärer Schreibweise (d. h. als Dualzahl) hintereinander und wandle die erhaltene Folge aus Einsen und Nullen wieder in eine normale Dezimalzahl um. Der Vorteil dieser Methode liegt zum einen darin, daß dadurch größere Zahlen entstehen (nicht nur Zahlen bis 255, denn der ASCII-Code besteht aus 255 Zeichen), des weiteren kann ich jetzt auch Sonderzeichen oder Satzzeichen verschlüsseln, was mir bei Verwendung des normalen Alphabetes mit nur 26 Buchstaben nicht möglich war.

Was heißt aber jetzt dieses ASCII? Dies ist eine amerikanische Abkürzung und bedeutet: **American Standard Code of Information Interchange**, also Amerikanische Standardcodierung zum Informationsaustausch. Sie ist so angelegt, daß sie allen Buchstaben, Zahlen, Satz- und Sonderzeichen, die zum Datenaustausch notwendig sind, eine Zahl zuweist, so daß auch Texte maschinenverständlich werden. Was heißt maschinenverständlich? Ein Computer

kennt eigentlich nur zwei Dinge: Strom oder kein Strom. Mit diesen beiden Zustände muß er nun alles tun können, was wir so von ihm verlangen. Da er aber eigentlich gar nichts von dem versteht, was wir wollen (ein Computer beherrscht keine Sprachen), muß alles in für ihn verständliche Information umgewandelt werden, d. h. also Strom oder kein Strom, 1 oder 0. Diese beiden Zustände lassen sich am besten durch Zahlen, die im Binärsystem geschrieben wurden, widerspiegeln. So hat im ASCII-Code der Buchstabe 'A' die Nummer '65' - in binärer Schreibweise '01000001'. Der Computer erhält diese Information, indem ihm in bestimmten Abständen Strom bzw. kein Strom geschickt wird, und jetzt weiß er, daß er es mit dem Buchstaben 'A' zu tun hat. Er unterscheidet natürlich auch Klein- und Großbuchstaben, so daß die Kodierung für 'a' anders aussieht als die für den entsprechenden Großbuchstaben.

Doch zurück zum asymmetrischen Kryptosystem. Zunächst sollen hier jedoch einmal ein paar mathematische Grundlagen geschaffen werden, damit später das System verstanden werden kann. Wichtig sind hier vor allen Dingen Berechnungen mit der bereits weiter oben erwähnten Modulo-Funktion. Sie liefert wie gesagt den ganzzahligen Rest einer Division. Im folgenden sollen einige Gesetze bewiesen werden, die bei der Berechnung mittels der Modulo-Funktion bedeutsam sind:

$$\begin{aligned} \mathbf{a} \bmod \mathbf{b} &= \mathbf{r}_1 \\ \mathbf{c} \bmod \mathbf{b} &= \mathbf{r}_2 \end{aligned}$$

$$\begin{aligned} a &= x \cdot b + r_1 \\ c &= y \cdot b + r_2 \\ \Rightarrow a + c &= x \cdot b + r_1 + y \cdot b + r_2 = b \cdot (x + y) + r_1 + r_2 \\ \Rightarrow & \\ & \mathbf{(a + c) \bmod b = (r_1 + r_2) \bmod b} \end{aligned}$$

Des weiteren gilt:

$$\begin{aligned} a \cdot c &= (x \cdot b + r_1) \cdot (y \cdot b + r_2) = xyb^2 + r_1yb + r_2xb + r_1 \cdot r_2 \\ \Rightarrow & \\ & \mathbf{(a \cdot c) \bmod b = (r_1 \cdot r_2) \bmod b} \end{aligned}$$

Zum dritten kann ich auch noch sagen:

$$\begin{aligned} a^k &= (xb + r_1)^k = x^k b^k + kx^{k-1}b^{k-1}r_1 + \dots + kxb r_1^{k-1} + r_1^k \\ \Rightarrow & \\ & \mathbf{a^k \bmod b = (a \bmod b)^k \bmod b = r_1^k \bmod b} \end{aligned}$$

Doch benötigen wir zum Verstehen des Kryptosystems, daß ich vorstellen möchte, noch etwas Zahlentheorie. Eine wichtige Rolle spielt beispielsweise der Satz von

Euler. Leonhard Euler war ein Schweizer Mathematiker und lebt von 1707 bis 1783. Auf ihn geht die Funktion $\varphi(n)$ zurück, die deshalb auch als Eulersche Funktion bezeichnet wird. Die Funktion $\varphi(n)$ (sprich: phi von n) liefert uns die Anzahl der zu n teilerfremden Zahlen (wobei n Element \mathbf{N}). Das heißt beispielsweise:

$$\varphi(10) = 4,$$

denn die Anzahl der Zahlen, die kleiner als 10 sind und mit dieser Zahl keinen Teiler gemeinsam haben, ist 4. (Es sind in unserem Fall die Zahlen 1, 3, 7, 9.)

Doch kommen wir jetzt zum Satz von Euler. Dieser mathematische Satz besagt folgendes:

Ich wähle zwei beliebige Zahlen a und b, die die Bedingung erfüllen müssen, daß sie teilerfremd sein sollen, d. h. $\text{ggT}(a, b) = 1$. Dann gilt:

$$\mathbf{a}^{\varphi(b)} \bmod \mathbf{b} = \mathbf{1}$$

Dies ist der Satz von Euler. Beweisen möchte ich diesen Satz doch in einer Sonderform. Angenommen, b sei irgendeine Primzahl p. Daraus folgt nun, daß $\varphi(p) = p-1$ ist, denn alle Zahlen von 1 bis p-1 sind zu p teilerfremd, da p ja eine Primzahl ist und somit keinen anderen Teiler als 1(, die ja zu allen Zahlen teilerfremd ist) und p hat. Wenn also der Satz von Euler gilt, so kann ich sagen, daß

$$\mathbf{a}^{\varphi(p)} \bmod \mathbf{p} = \mathbf{a}^{p-1} \bmod \mathbf{p} = \mathbf{1}.$$

Da $\mathbf{a} \bmod \mathbf{p} = \mathbf{a}$ (für $a < p$), kann ich unter Zuhilfenahme des oben aufgestellten Gesetzes für die Modulo-Funktion sagen

$$(\mathbf{a} \bmod \mathbf{p}) \cdot (\mathbf{a}^{p-1} \bmod \mathbf{p}) = (\mathbf{a} \cdot \mathbf{a}^{p-1}) \bmod \mathbf{p} = \mathbf{a}^p \bmod \mathbf{p} = \mathbf{a}.$$

Ich möchte nun folgende Aussage beweisen:

$$\mathbf{a}^p \bmod \mathbf{p} = \mathbf{a} \bmod \mathbf{p}$$

Für diesen Beweis nutze ich die vollständige Induktion. Diese Beweismethode geht nun so vor, daß ich zunächst überprüfe, ob die Aussage für $a=1$ gilt. Jetzt gehe davon aus, daß die Aussage auch für $a = k$ gilt und prüfe, ob sie dann auch für $a = k + 1$ gilt. Denn wenn ich weiß, daß eine Aussage für $a=1$ und auch für jede folgende Zahl gilt, so gilt die Aussage auch für $a=2, 3, 4, \dots$ usw.

Induktionsanfang: $1^p \bmod p = 1 \bmod p$ ist richtig.

Induktionsschritt:

Annahme: $k^p \bmod p = k \bmod p$

zu zeigen: $(k+1)^p \bmod p = (k+1) \bmod p$

$(k+1)^p$ läßt sich auch schreiben als:

$$(k+1)^p = k^p + a \cdot k^{p-1} + b \cdot k^{p-2} + \dots + b \cdot k^2 + a \cdot k + 1$$

Aber wie sehen die Faktoren a, b usw. aus? Diese sogenannten Binomialkoeffizienten sehen nun folgendermaßen aus:

Faktor f an der i. Stelle: $f = \binom{p}{i-1}$ (sprich: p über i-1)

(i sei dabei die Nummer des Summanden, wenn ich den ersten mitzähle).

Doch was heißt dieses $\binom{p}{i-1}$? Dies möchte ich an einem Beispiel verdeutlichen:

$$\binom{5}{2} = \frac{5 \cdot 4}{1 \cdot 2} = 10 \quad \text{oder} \quad \binom{7}{3} = \frac{7 \cdot 6 \cdot 5}{1 \cdot 2 \cdot 3} = 35$$

Ganz allgemein bedeutet das:

$$\binom{p}{k} = \frac{p \cdot (p-1) \cdot (p-2) \cdot \dots \cdot (p-k+1)}{1 \cdot 2 \cdot 3 \cdot \dots \cdot k}$$

Somit heißt unsere Formel von oben:

$$\binom{p}{0} \cdot k^p + \binom{p}{1} \cdot k^{p-1} + \binom{p}{2} \cdot k^{p-2} + \dots + \binom{p}{p-2} \cdot k^2 + \binom{p}{p-1} \cdot k + \binom{p}{p} \cdot 1$$

* $\binom{p}{0}$ ist definiert als 1

Da Binomialkoeffizienten natürliche Zahlen sein müssen und p eine Primzahl ist, heißt das, daß in allen Faktoren, außer im ersten und im letzten, der Faktor p vorkommt, denn er läßt sich nicht wegkürzen. Daraus folgt nun:

$$(k+1)^p \bmod p = (k^p + 1) \bmod p = k^p \bmod p + 1 \bmod p = k \bmod p + 1 \bmod p = (k+1) \bmod p$$

* Siehe Annahme

$$a^p = k \cdot p + a \quad | -a$$

$$a^p - a = k \cdot p$$

$$a \cdot (a^{p-1} - 1) = k \cdot p$$

Da $\text{ggT}(p, a) = 1 \Rightarrow \text{ggT}(k, a) = a \Rightarrow \frac{k}{a}$ ist eine natürliche Zahl

$$a^{p-1} - 1 = \frac{k}{a} \cdot p \quad | +1$$

$$a^{p-1} = \frac{k}{a} \cdot p + 1 \quad | \text{mod } p$$

$$a^{p-1} \text{ mod } p = 1$$

Doch genug der Mathematik, kommen wir nun endlich zu unserem asymmetrischen Kryptosystem.

Dieses Kryptosystem wurde von Ronald Rivest, Adi Shamir und Leonard Adleman entwickelt, weshalb es auch mit den Anfangsbuchstaben ihrer Nachnamen als RSA-Algorithmus bezeichnet wurde, und sieht nun folgendermaßen aus: Wir wählen zwei möglichst große Primzahlen p und q , bilden das Produkt daraus und erhalten n . Dann wählen wir einen öffentlichen Schlüssel e , mit dem wir unseren Klartext verschlüsseln können. Doch benötigen wir noch einen zweiten, geheimen Schlüssel d , mit dem der Geheimtext wieder entschlüsselt werden kann. Zwischen e , d und n besteht folgender Zusammenhang:

$$\text{ggT}(e \cdot d, \varphi(n)) = 1$$

Da n ein Produkt der Primzahlen p und q ist, haben nur folgende Zahlen Teiler mit n gemeinsam:

$$1 \cdot p, 2 \cdot p \dots q \cdot p : \text{insgesamt } q \text{ Zahlen}$$

$$1 \cdot q, 2 \cdot q \dots p \cdot q : \text{insgesamt } p \text{ Zahlen}$$

↓

$$\varphi(n) = p \cdot q - p - q + 1 = p \cdot (q-1) - q + 1 = p \cdot (q-1) - (q-1) = (p-1) \cdot (q-1)$$

Da der $\text{ggT}(e \cdot d, \varphi(n)) = 1$ sein soll, müssen beide Zahlen e und d teilerfremd zu $\varphi(n)$ sein. Dazu wähle ich zunächst eine Zahl e , von der ich weiß, daß sie zu $\varphi(n)$ teilerfremd ist und berechne die zweite Zahl d mit Hilfe des euklidischen Algorithmus, der uns normalerweise den größten gemeinsamen Teiler zweier Zahlen liefert.

Doch machen wir auch hier ein Beispiel:

$$p = 5$$

$$q = 11$$

$$n = p \cdot q = 5 \cdot 11 = 55$$

$$\varphi(n) = (5-1) \cdot (11-1) = 4 \cdot 10 = 40$$

$$e=7 \text{ (willkürlich festgelegt)}$$

$$\begin{aligned}
 40 &= 5 \cdot 7 + 5 \\
 7 &= 1 \cdot 5 + 2 \\
 5 &= 2 \cdot 2 + 1 \\
 &\Downarrow \\
 1 &= 5 - 2 \cdot 2 \\
 1 &= 5 - 2 \cdot (7 - 1 \cdot 5) \\
 1 &= 3 \cdot 5 - 2 \cdot 7 \\
 1 &= 3 \cdot (40 - 5 \cdot 7) - 2 \cdot 7 \\
 1 &= 3 \cdot 40 - 15 \cdot 7 - 2 \cdot 7 \\
 1 &= 3 \cdot 40 - 17 \cdot 7 \quad | + 40 \cdot 7 - 40 \cdot 7 \\
 1 &= 3 \cdot 40 - 7 \cdot 40 - 17 \cdot 7 + 40 \cdot 7 \\
 1 &= 40 \cdot (3 - 7) + 7 \cdot (40 - 17) \\
 1 &= -4 \cdot 40 + 7 \cdot 23 \\
 &\Downarrow \\
 d &= 23
 \end{aligned}$$

Wie wird nun mit diesem Kryptosystem verschlüsselt? Eigentlich ziemlich einfach:

Den Geheimtext c erhalte ich, indem ich $c = m^e \bmod n$ rechne. Den Klartext m' erhalte ich wieder, wenn ich $m' = c^d \bmod n$ rechne. Jetzt wäre nur noch zu beweisen, daß $m' = m$ ist. Versuchen wir es einmal mit unserem Beispiel:

$$\begin{aligned}
 m &= 8 \\
 c &= m^e \bmod n = 8^7 \bmod 55 = 2 \\
 m' &= c^d \bmod n = 2^{23} \bmod 55 = 8
 \end{aligned}$$

Und siehe da, es funktioniert. (Beim Nachrechnen bitte darauf achten, daß auch ein Taschenrechner nur ein Taschenrechner ist und manchmal doch etwas ungenau rechnet - also bitte mal ein Auge zudrücken, wenn er nicht genau die oben angegebenen Zahlen ausspuckt, sondern aufgrund seiner Rundungsfehler leicht abweichende Werte ausgibt.) Aber warum funktioniert diese Methode? Wie gesagt, ich muß beweisen, daß $m' = m$ ist. Aber was ist m' ?

$$m' = (m^e \bmod n)^d = m^{e \cdot d} \bmod n$$

Schreiben wir für $e \cdot d = 1 + k \cdot (p-1) \cdot (q-1)$ so erhalten wir:

$$m' = m^{1+k \cdot (p-1) \cdot (q-1)} \bmod n$$

Doch gehen wir in unserem Beweis zunächst schrittweise vor und rechnen:

$$m^{1+k \cdot (p-1) \cdot (q-1)} \bmod p = m \cdot m^{k \cdot (p-1) \cdot (q-1)} \bmod p = m \cdot (m^{p-1})^{k \cdot (q-1)} \bmod p$$

$$= m \text{ mod } p \cdot (m^{p-1} \text{ mod } p)^{k \cdot (q-1)} \text{ mod } p$$

da $\text{ggT}(m, p)$ mit größter Wahrscheinlichkeit 1 ist - weil p ja eine Primzahl ist, also müßte $m = p$ oder m ein Vielfaches von p sein, damit dies nicht zutrifft - können wir jetzt den oben bewiesenen Satz von Euler anwenden und erhalten:

$$m \text{ mod } p \cdot 1^{k \cdot (q-1)} \text{ mod } p = m \text{ mod } p$$

Den gleichen Schritt führe ich mit q durch:

$$\begin{aligned} m^{1+k \cdot (p-1) \cdot (q-1)} \text{ mod } q &= m \cdot m^{k \cdot (p-1) \cdot (q-1)} \text{ mod } q = m \cdot (m^{q-1})^{k \cdot (p-1)} \text{ mod } q \\ &= m \text{ mod } q \cdot (m^{q-1} \text{ mod } q)^{k \cdot (p-1)} \text{ mod } q \end{aligned}$$

auch hier gilt das gleiche wie für p , daraus folgt nun nach dem Satz von Euler:

$$m \text{ mod } q \cdot 1^{k \cdot (p-1)} \text{ mod } q = m \text{ mod } q$$

Wie wir gesehen haben, ergibt sich in beiden Fällen ein Rest von m (für $m < q$ bzw. $m < p$, was ja sehr wahrscheinlich ist, den p und q sollen möglichst große Primzahlen sein). Ziehe ich nun von $m^{e \cdot d}$ einmal m ab, so erhalte ich im ersten Falle eine Zahl, die durch q teilbar ist, im zweiten Fall ist diese Zahl auch durch q teilbar (, was übrigens auch gilt, wenn m nicht kleiner als p oder q ist). Das heißt also, daß diese Zahl auch durch $p \cdot q = n$ teilbar sein muß, es gilt:

$$m^{e \cdot d} - m \text{ mod } n = 0$$

addiere ich nun wieder $m \text{ mod } n = m$ (m ist auf jeden Fall kleiner als n), so erhalte ich:

$$m^{e \cdot d} - m \text{ mod } n + m \text{ mod } n = m^{e \cdot d} \text{ mod } n = 0 + m = m$$

damit haben wir also bewiesen, daß gilt:

$$m' = m^{e \cdot d} \text{ mod } n = m$$

Aber warum soll dieses Kryptosystem nun so sicher sein? Ich müßte mit den Kenntnissen, die öffentlich zugänglich sind, auf den geheimen Schlüssel d schließen können. Mit einer Häufigkeitsanalyse des Geheimtextes kommen wir nicht weiter, da ja nicht jeder Buchstabe einzeln, sondern immer mehrere, aufeinanderfolgende Buchstaben verschlüsselt werden. Und selbst wenn jeder Buchstabe einzeln verschlüsselt würde, so würde uns die Kenntnis über die

Häufigkeit der einzelnen Buchstaben nicht weiterhelfen, denn bei unserer Verschlüsselungsmethode werden ja nicht nur die Buchstaben des normalen Alphabets, sondern die Zeichen des gesamten ASCII-Codes verschlüsselt, für die es wohl kaum eine allgemeine Häufigkeitsanalyse geben kann. Doch da besteht doch noch ein Zusammenhang zwischen n , e (die öffentlich zugänglich sein sollen) und d (die einzige Zahl, die geheim ist).

Das Problem an der Sache ist nämlich folgendes: Zum Berechnen von d benötigen wir $\varphi(n) = (p-1) \cdot (q-1)$. Ich müßte n also in seine beiden Primfaktoren p und q zerlegen können. Da p und q zwei sehr, sehr große Primzahlen sind, müßten man n durch alle Zahlen bis zu der kleineren der beiden teilen. Auch wenn man natürlich die ganze Reihe von kleinen, bekannten Primzahlen und deren Vielfache wegfallen lassen könnte, so ist die Anzahl der durchzuführenden Divisionen doch noch groß genug, um selbst mit den heutigen Computern nicht ohne weiteres gelöst werden zu können. Bisher ist nämlich noch kein effektiverer Algorithmus gefunden worden, der eine Zahl zuverlässig und schnell in seine Primfaktoren zerlegt.

Aber man konnte noch nicht beweisen, daß es keinen solchen Algorithmus gibt, und wer auf die Vermutung baut, daß die Mathematiker keine leichtere Methode finden werden, der glaubt zwar an die Sicherheit des RSA-Algorithmus, doch reine Vermutungen werden oft sehr schnell widerlegt. Wenn man es sogar noch eine Spur genauer nimmt, so beleidigt jeder die Mathematik, wenn er behauptet, daß der RSA sicher sei, weil es keine sichere und schnelle Methode gibt, mit der sich eine sehr große Zahl in ihre Primfaktoren zerlegen läßt.

Schlußbemerkung

Wir haben jetzt einige Methoden zum Chiffrieren kennengelernt, aber auch andere, um diese Chiffrierungen wieder zu knacken. In einem Fall war sehr viel Mathematik notwendig, um den Algorithmus zu verstehen, in einem anderen Fall haben wir zur Analyse des Geheimtextes sehr viel Mathematik benötigt. In jedem Fall sind die Kryptologie und die Mathematik zwei Wissenschaften, die nicht ohne weiteres getrennt werden können - ja ohne Mathematik wäre Kryptologie gar nicht möglich.

Ich hoffe, daß es mir gelungen ist, Ihnen/Euch einen verständlichen Einblick in die Wissenschaft des Verheimlichens gegeben zu haben, und schließe nun diese Facharbeit mit einem Zitat von Erich Kästner: *”Manche Menschen benützen ihre Intelligenz zum Vereinfachen, manche zum Komplizieren”*.

Quelle

Beutelspacher, Albrecht

Kryptologie

Braunschweig/Wiesbaden [Vieweg] ³1993

Zudem wurde diese Facharbeit mit freundlicher Unterstützung von Herrn Bernd von den Hoff angefertigt, der mir mit seinem Fachwissen bei der Lösung einiger Probleme behilflich war.