DER DES ALGORITHMUS

ALEXEJ ROTAR & SIMON STERNSDORF[1]

30.3.2016

CONTENTS

LIST OF FIGURES

1 EINLEITUNG

Verschlüsselung fand schon im alten Rom Anwendung. Einer der ersten und wohl auch bekanntesten Verschlüsselungsalgorithmen der Werlt, die sogeannte "Cäsar-Chiffre", findet man heutzutage in Popkultur um die ganze Welt. Es ist ein sehr einfaches symmetrisches Verfahren, bei dem das uns bekannte Alphabet zum Einsatz kommmt. Jeder Buchstabe wird durch einen anderen Buchstaben aus unserem Alphabet ersetzt, der um eine bestimmte Anzahl weiter hinten in der Reihenfolge steht. Das Ganze ist zyklisch und ergibt am Ende eine einfache Umwandlungstabelle. [1] Dieser Algorithmus ist heutzutage natürlich viel zu leicht zu knacken mit gerade einmal 25 verschiedenen Verschlüsselungsmöglichkeiten. Aber das Prinzip des symmetrischen Verschlüsselns mittels einer Chiffre ist uns bis heute erhalten geblieben. Der hier behandelte Algorithmus arbeitet im Grunde ganz ähnlich: Der DES Algorithmus.

1.1 Geschichte

Der DES Algorithmus ist trotz seiner Sicherheitsprobleme immer noch einer der weltweit am weitesten verbreiteten Verschlüsselungsalgorithmen. Nicht umsonst heißt er Data Encryption Standart. Vor allem in dem Derivat "triple-DES" oder auch verkürzt "3DES" wird er noch eingesetzt und wird es wohl auch noch viele Jahre lang werden. Er wurde 1974 in Folge einer Ausschreibung des NBS, des National Bureau of Standars, für einen sicheren Verschlüsselungsstandart zum Verschicken von Daten in den USA von IBM eingereicht. IBM beschäftigte damals unter anderem Horst Feistel, auf dessen Chiffren später genauer eingegangen wird. DES basiert grob auf dem schon zuvor entwickelten Algorithmus Luciver.[2] Die Verschlüsselung wurde 1976 nach Anpassungen der NSA, der National Security Agency, als allgemeiner Standart für verschlüsselte Datenverbindungen im Internet übernommen, und wurde vor allem von der Bankenindustrie und der US-Amerikanischen Regierung für die Kommunikation eingesetzt. [3]

1.2 Kritik an Sicherheit

Die Anpassungen der NSA sind sehr umstritten. So soll etwa der Sicherheitsgrad gesenkt worden sein durch Verkürzung der Schlüssellänge von 128 Bit auf 56 Bit und eventuell sogar die zur Verschlüsselung notwendigen Substitutions-Boxen von der NSA verändert worden sein.Man soll versucht haben sich eine Hintertür, einen sogenannten "backdoor" in den Algorithmus einzubauen. Da DES von Anfang an nur für den normalen Datenverkehr, aber nicht für Dokumente der höchsten Sicherheitsstufe eingesetzt

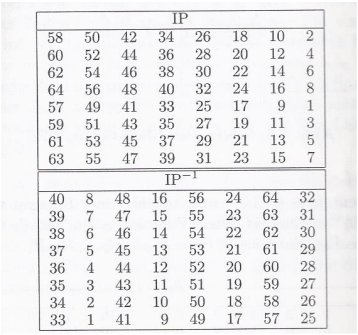
1 http://www.kryptowissen.de/caesar-chiffre.html - 07.03.2016 13:50
2 http://page.math.tu-berlin.de/~kant/teaching/hess/krypto-ws2006/des.htm - 07.03.2016 13:50
3 http://page.math.tu-berlin.de/~kant/teaching/hess/krypto-ws2006/des.htm - 07.03.2016 13:50

wurde, ist es bis heute sehr umstritten wie sicher DES wirklich ist.[4] Allerdings haben sich die meisten dieser Spekulationen nicht bestätigt.[5]

2 MATHEMATISCHE GRUNDLAGEN

2.1 Permutation und Expansion

Der DES-Algorithmus arbeitet mit sehr einfachen mathematischen Methoden. Der Schlüssel ist hierbei immer 64 Bit lang, wobei immer das letzte von 8 Bit als Korrektur-Bit verwendet wird, mit dem Speicher- und Übertragungsfehler ausgeglichen werden können.[6] Dieser Schlüssel wird so zum Verschlüsseln auf die Nachricht angewandt, die jeweils in 64 Bit Blöcke unterteilt wird. Eine Permutation auf diese Bits entspricht einem Verschieben der Bits auf eine neue Position innerhalb des Blocks. Das Ganze wird gerne als Tabelle dargestellt, als sogenannte Permutations-Tabelle.



IP

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

IP^{-1}

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Figure 1: Eingangs-Permuations Tabelle[7]

Jede Zahl beziffert genau die neue Position des Bits innerhalb des Permutatins-Blocks. So wird hier das erste Bit auf das 58te abgebildet, das zweite auf das 50te und so weiter. Nach dem selben Prizip wird dann die Expansion angewandt, allerdings ist dabei die Tabelle größer als der ursprüngliche Block. Das heißt hier werden manche Bits auf mehrere Stellen gesetzt.

2.2 Substitution

Weiter wichtig sind zudem Substitutionen. Auf ihren Nutzen für die Sicherheit des Verfahrens wird später noch genauer eingegangen. Im Allgemeinen wird bei der Substitution ein Block an Bits durch einen anderen ersetzt. Dies geschieht wiederum durch Tabellen, wobei man hier beim DES auch von

4 http://www-lehre.informatik.uni-osnabrueck.de/~rspier/referat/internet/DES-Algorithmus.html - 07.03.2016 13:50
5 Christian Karpfinger, Hubert Kiechle. Kryptologie. 1. Auflage. Seite 44. Vieweg+Teubner, 2010
6 Buchmann. Einführung in die Kryptologie. 4te, erweiterte Auflage. Seite 104. Springer-Verlag Berlin Heidelberg, 2008
7 Buchmann. Einführung in die Kryptologie. 4te, erweiterte Auflage. Seite 105. Springer-Verlag Berlin Heidelberg, 2008

einer gleichzeitigen Kontraktion sprechen kann, da die Bitfolge verkürzt wird. [8]

Expansion und Permutation sind lineare Abbildungen, wobei die Permutation eine Abbildung $\mathbb{Z}_2^{32} \to \mathbb{Z}_2^{32}$ und die Expansion eine Abbildung E: $\mathbb{Z}_2^{32} \to \mathbb{Z}_2^{48}$sind. Die Substitution ist eine nicht-lineare Abbildung.[9]

2.3 Kryptosysteme

Wir verwenden in dieser Arbeit die Definition eines Kryptosystems wie folgt: Ein Kryptosystem ist ein Tupel(P, C, K, f, g) wobei P, C und K nicht leer sein dürfen. Hierbei ist:

- **P** die Klartextmenge, sprich der Text der verschlüsselt werden soll
- **C** die Geheimtextmenge, sprich der verschlüsselte Text
- **K** die Schlüsselmenge, also der Schlüssel der zum verschlüsseln der Klartextmenge zur Geheimtextmenge verwendet wurde
- **f** die Verschlüsselungsfunktion, eine Abbildung $f : P \times K \to C$, die aus der Klartextmenge die Geheimtextmenge bildet
- **g** die Entschlüsselungsfunktion, eine Abbildung $g : C \times K \to P$, die aus der Geheimtextmenge die Klartextmenge bildet

Weiter muss gelten das $\forall k \in K : \exists k' \in K : g_{k'} \circ f_k = id_P$, was gleichbedeutent ist mit Injektivität und Surjektivität der Abbildung. Ohne diese Vorraussetzung wäre eine verschlüsselte Nachricht nicht mehr entschlüsselbar. Bei DES gilt $k' = k$. [10]

3 PERFEKTE SICHERHEIT

3.1 Konzept

Um zu verstehen wie sicher oder unsicher der DES-Algorithmus ist, braucht man das Konzept der perfekten Sicherheit.
Hierbei gehen wir davon aus, dass der Angreifer über unendliche Rechenkapazitäten verfügt. Weiterhin haben wir ein Kryptosystem $\pi = (P, C, K, f, g)$, das ein nach Kerckhoff's Prinzip sicheres Verschlüsselungsverfahren enthält, was heißt f und g sind allgemein bekannt.[11] Wir haben nun Wahrscheinlichkeits-Verteilungen Ws auf P,C und K, zudem sind P, C und K die Zufallsvariablen für die Ws-Verteilungen. Nun ziehen wir ein $p \in P$ mit $Ws[P = p]$ und analog für C und K. Die Wahrscheinlichkeit für $P = p$ und $C = c$ sollen dabei > 0 sein. Nun ist unser Kryptosystem und damit unser Verschlüsselungsverfahren sicher wenn der folgende Satz gilt:

8 Buchmann. Einführung in die Kryptologie. 4te, erweiterte Auflage. Seite 107. Springer-Verlag Berlin Heidelberg, 2008
9 Christian Karpfinger, Hubert Kiechle. Kryptologie. 1. Auflage. Seite 52. Vieweg+Teubner, 2010
10 Christian Karpfinger, Hubert Kiechle. Kryptologie. 1. Auflage. Seite 9. Vieweg+Teubner, 2010
11 http://www.cits.rub.de/imperia/md/content/may/ws1516/krypto_i.pdf - 09.03.2016 15:05

Satz 1 (Chiffretext-Verteilung).

$$Ws[P = p|C = c] = Ws[P = p]|\forall p \in P, c \in C$$

Proof. Annahme: π sei perfekt sicher. Dann gilt nach Satz von Bayes:

$$\frac{Ws[P = p|C = c] * Ws[P = p]}{Ws[C = c]} = Ws[P = p|C = c] = Ws[P = p]$$

Daraus folgt

$$Ws[P = p|C = c] = Ws[C = c]$$

Aus $Ws[P = p|C = c] = Ws[C = c]$ folgt mit dem Satz von Bayes:

$$Ws[P = p] = Ws[P = p|C = c]$$

Damit ist nachgewiesen, dass π perfekt sicher ist. □

Der für diesen Satz nötige Satz lautet wie folgt:

Satz 2 (Satz von Bayes).
Für zwei Ereignisse A und B, wobei $B \neq 0$:

$$P_B(A) = \frac{P(A) * P_A(B)}{P(B)}$$

Hierbei bezeichnet $P_B(A)$ die Wahrscheinlichkeit des Ereignisses A unter Vorraussetzung des Eintretens von Bedingung B. Equivalent für $P_A(B)$ $P(A)$ ist die sogenante Anfangswahrscheinlichkeit für das Ereignis A. Sie meint, dass das Ereignis A unabhängig zu betrachten ist. [12]

[13] Ein Angreifer der sowohl die entschlüsselte Nachricht wie auch die verschlüsselte besitzt hat mit diesem System keine Vorteile.[14] Man kann auch sagen p und c sind stochastisch unabhängig. Mit dem Satz der Chiffren-Verteilung kann man folgendes nachweisen:
Satz 3 (Ununterscheidbarkeit von Verschlüsselung). Ein Verschlüsselungsverfahren π ist perfekt sicher wenn gilt:

$$p_0, p_1 \in P, c \in C : Ws[P = p_0|C = c] = Ws[P = p_1|C = c]$$

Proof. Aus Satz 1 folgt für ein π das perfekt sicher ist:

$$Ws[P = p_0|C = c] = Ws[C = c]$$

$$\exists c \in C : Ws[C = c] = Ws[P = p_1|C = c]$$

Sei $p' \in P$ frei wählbar. Dann gilt:

$$Ws[C = c] = \sum_{p \in P} Ws[P = p|C = c] * Ws[P = p]$$

$$= Ws[P = p'|C = c] * \sum_{p \in P} Ws[P = p]$$

$$= Ws[P = p'|C = c]$$

Insgesamt folgt daraus die perfekte Sicherheit von π [15] □

12 http://matheguru.com/stochastik/36-satz-von-bayes.html - 25.03.2016 15:50
13 http://www.mathebibel.de/satz-von-bayes - 25.03.2016 14:20
14 http://www.cits.rub.de/imperia/md/content/may/0910/ws0910/krypto1ws09/02_perfekt.pdf-Seite 5 - 09.03.2016 15:30
15 http://www.cits.rub.de/imperia/md/content/may/0910/ws0910/krypto1ws09/02_perfekt.pdf - Seite 6 - 09.03.2016 16:20

Man kann außerdem nachweisen, dass für das perfekte Verschlüsselungsverfahren gelten muss:
Satz 4 (Minimale Größe des Schlüsselraumes). Annahme: π sei perfekt sicher. Dann gilt: $|K| \geqslant |P|$

Proof. Beweis durch Widerspruch: Annahme: $|K| < |P|$ Für ein $c \in C$ definieren wir ein $G(c) = p \mid p = g(c,k)$ mit $k \in K$. Es muss $|G(c)| \leqslant |K|$ gelten, da ein Schlüssel k genau einen Klartext p liefert. Es gilt aber $|K| < |P|$. Daraus folgt das $|G(c)| < |P|$. Somit muss es ein $p \in P$ geben für das gilt: $Ws[P = p|C = c] = 0 < Ws[P = p]$. Das würde bedeuten, dass π nicht perfekt sicher sein kann, da die Ws als > 0 definiert wurden. [16] □

Mit dem Satz von Shannon können wir uns zudem noch die Verteilung der Schlüssel anschauen.
Satz 5 (Shannon). Für ein Kryptosystem π = (P,C,K,f,g) mit $|P| = |C| = |K|$ gilt: π ist perfekt sicher gdw : alle $k \in K$ werden durch f gleichverteilt gewählt mit einer Wahrscheinlichkeit von $\frac{1}{|K|}$. Zudem gibt es für alle $p \in P, c \in C$ genau ein $k \in K$ sodass gilt: c = f(p,k).

Hier beweisen wir zunächst die Gegenrichtung($\Leftarrow$):

Proof. Wir können ein $c \in C$ genau zu einem $p \in P$ entschlüsseln mit einem $k \in K$, bedeutet g(c,k) = p. Dies geschieht mit $Ws[\frac{1}{|K|}]$ gleichverteilt. Wir folgern:

$$Ws[C = c|P = p] = \frac{1}{|K|}$$

für alle $p \in P$ Hieraus folgt mit Satz 2:

$$Ws[C = c|P = p_0] = \frac{1}{|K|} = Ws[C = c|P = p_1]$$

Die Hinrichtung ($\Rightarrow$) folgt somit: Widerspruchsbeweis: Wir nehmen an: $\exists(p, c)$ mit $c \neq f(p, k)$ für alle $k \in K$ Deswegen gilt: $Ws[P = p|C = c] = 0 < Ws[P = p]$. Dies steht im Widerspruch zu unserer Definition von einem perfekt sicherem System.

Nehmen wir anders herum an: $\exists(p, c)$ mit $c = f(p, k)$ für mehrere $k \in K(\geqslant 2)$. Dann gilt: $\exists(p', c')$ mit $c' \neq f(p', k)$ für alle $k \in K$.

Dies erzeugt einen Widerspruch wie oben. Daraus folgern wir: Es gibt für jedes feste $p \in P, c \in C$ genau ein $k \in K$ für das gilt: f(p,k) = c. Im weiteren ergibt sich für alle p, p' : $Ws[K = k_p] = Ws[C = c|P = p] = Ws[C = c|P = p'] = Ws[K = k'_p] \rightarrow Ws[K = k] = \frac{1}{|K|}$ für alle $k \in K$[17] □

Intuitiv bedeutet perfekte Sicherheit also:
Der Schlüssel, der zur Verschlüsselung der Nachricht verwendet wird muss mindestens so lang sein wie die Nachricht selbst. Der Schlüssl muss zudem noch perfekt gleichverteilt von der Verschlüsselungsfunktion gewählt werden. Der Angreifer hat keinerlei Vorteile durch das Wissen um ein Klartext - verschlüsselter Text Paar. Das schließt auch sogenannte Meet-In-The-Middle

16 http://www.cits.rub.de/imperia/md/content/may/0910/ws0910/krypto1ws09/02_perfekt.pdf - Seite 8 - 09.03.2016 16:20
17 http://www.cits.rub.de/imperia/md/content/may/0910/ws0910/krypto1ws09/02_perfekt.pdf - Seite 9 - 09.03.2016 16:20

Attacken aus, auf die später noch eingegangen wird. Mit all diesen nicht gerade leicht zu erfüllenden Bedingungen gibt es nicht viele Beispiele aus der Vergangenheit, von Verschlüsselungsverfahren die die Kriterien der perfekten Sicherheit erfüllten.

3.2 Vernam'sches One-Time-Pad

Eines dieser wenigen Beispiel ist das Vernam'sche One-Time-Pad, eine 1918 von Gilbert Vernam entwickelte Verschlüsselungsmethode, die zum verschlüsseln des roten Telefons im Kalten Krieg zwischen dem US-Präsidenten und dem sowjetischen Generalsekretär genutzt wurde. Die Besonderheit ist, dass der Schlüssel genauso lang wie die zu verschlüsselnde Nachricht war. Zur Ver-und Entschlüsselung wurden Buchstaben nach einer Tabelle addiert.

+	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
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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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

Figure 2: One-Time Pad Tabelle[18]

Addiert wurde mit einer zufälligen Buchstabenfolge die beiden Seiten bekannt war.[19] Mathematisch wurde das One-Time-Pad definiert durch ein Kryptosystem $\pi = (P, C, K, f, g)$ wobei $P = C = K = \{0,1\}^l$ für das gilt:

$$\exists k \in K, \exists p \in P : c \in C = f(p,k) = p \oplus k$$

$$\exists k \in K, \exists c \in C : p \in P = g(c,k) = c \oplus k$$

h Annahme: Das One-Time-Pad ist ein perfekt sicheres Verfahren.

Proof. Da $C = P \oplus K$ gilt: $\forall p_0, p_1 \in Pundc \in C : Ws[C = c|P = p_0] = Ws[P \oplus K = p_0] = Ws[K = p_0 \oplus c] = \frac{1}{2^l} = Ws[C = c|P = p_1]$ □

Dies entspicht dem Satz über Ununterscheidbarkeit von Verschlüsselung. Somit ist das One-Time-Pad perfekt sicher. [20]
Das Beispiel zeigt gut, wie wenig praktikabel eine Verschlüsselung mit perfekter Verschlüsselung im Alltag ist. Ein so langer Schlüssel macht nicht nur das Ver- und Entschlüsseln langsam, sondern fördert auch das Auftreten von Fehlern. Wenn beim One-Time-Pad die zufälligen Buchstabenfolgen nicht übereinstimmten oder etwa falsch gelesen wurden, konnten beide Seiten nicht mehr kommunizieren und es mussten neue Folgen ausgetauscht

18 http://www.mathe.tu-freiberg.de/~hebisch/Praktikum11-4/Seite2.html - 09.03.2016 16:00
19 http://www.mathe.tu-freiberg.de/~hebisch/Praktikum11-4/Seite2.html - 09.03.2016 - 16:00
20 http://www.cits.rub.de/imperia/md/content/may/0910/ws0910/krypto1ws09/02_perfekt.pdf - Seite 7 - 09.03.2016 16:20

werden, was wiederum anfällig für Fehler und Angriffe von 3ten Parteien war. Somit kann das Ziel bei der Konstruktion eines Verschlüsselungsalgorithmus nie sein, ihn perfekt sicher zu machen, sondern nur sicher genug. In der Realität haben die Angreifer auf eine Verschlüsselung nie unbegrenzte Resourcen zur Verfügung. Somit ist ein Schlüssel, der sehr viel kürzer als die Länge des Klartextes ist, oftmals ausreichend. Auch ist die Wahl eines Algorithmus, der die Schlüssel absolut gleichverteilt wählt, sehr schwierig und nur bei einem riesigem Schlüsselraum realistsisch. Ein solches Verschlüsselungsverfahren zu bauen und zu verwalten ist nicht praktisch.

4 SCHLUSS

4.1 Kritik an DES

Schon recht früh nach Einführung des DES-Algorithmus als Standard-Verschlüsselung in vielen Bereichen des Öffenlichen Lebens kam Kritk auf. So wurde bereits früh, im Jahr 1975, vor der Möglichkeit eines Brute-Force Angriffs gewarnt, der aufgrund der recht kurzen Schlüssellänge gut möglich wäre. Im speziellen Martin Hellman und Whitfield Diffie, die Erfinder des Konzepts der Public-Key Verschlüsselung, warnten vor möglichen Angriffen von Regierungsorganisationen, im speziellen Geheimdienste. Diese hätten Zugriff auf die für damalige Verhältnisse recht hohe Rechenleistung.[21]

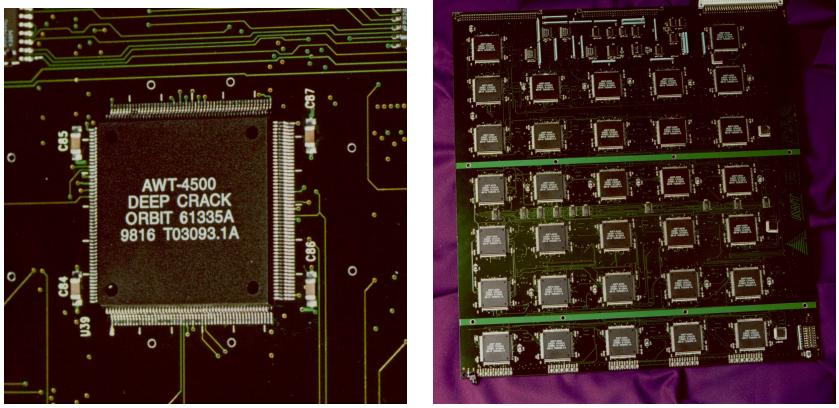
4.2 Brute-Force

Ein Brute-Force-Angriff ist die einfachste und gewaltsamste Art eine Verschlüsselung zu knacken. Dabei wird von einem Computer systematisch jede Kombination aus Zahlen, Buchstaben und Sonderzeichen bis zu einer zuvor festgelegten Länge durchprobiert. Je kürzer dabei die Länge des verwendeten Schlüssels ist desto schneller ist das Verfahren. Zudem hängt es von der verwendeten Rechenpower des Angreifers ab. Ein höherer Einsatz von Rechenleistung erhöht die Geschwindigkeit der Suche.[22]
Moderne Verschlüsselungen setzen deswegen heutzutage auf Schlüssel, die sehr viel länger sind als die 56 Bit des DES-Algorithmus. Der weit verbreitete Verschlüsselungsalgorithmus AES setzt auf Schlüssellängen bis zu 256 Bit. [23]

4.3 Der DES-Cracker

Im Jahr 1998 gelang es schließlich der Electronic Frontier Foundation, eine Organisation die sich für digitale Grundrechte und Privatssphäre einsetzt, mithilfe einer selbst gebauten "Crackers" den DES-Algorithmus mittels Brute-Force zu knacken. Die Maschine kostete nur 250.000 Dollar und benötigte

21 http://page.math.tu-berlin.de/~kant/teaching/hess/krypto-ws2006/des.htm - 11.3.2016 - 17:20
22 http://www.itwissen.info/definition/lexikon/Brute-Force-Angriff-brute-force-attack.html - 11.03.2016 17:20
23 https://www.boxcryptor.com/de/verschl - 11.03.2016 - 17:30



(a) Der verwendete Cracker-Chip. **(b)** Ein Chip-Board

Figure 3: DES-Cracker Chip [24]

nur 3 Tage um diese Aufgabe zu bewerkstelligen. Die EFF wollte damit der Öffentlichkeit aufzeigen, wie leicht der immer noch in weiten Teilen der Industrie eingesetzte Verschlüsselungsalgorithmus zu knacken war. 1999 gelang es mit dem DES-Cracker und einem weltweitem Computernetzwerk namens Distributed.Net eine DES-verschlüsselte Nachricht in nur 22 Stunden zu knacken. Die Computer schafften es zusammen auf eine Geschwindigkeit von 245 Millarden Schlüsseltests pro Sekunde. Das ganze geschah im Rahmen eines Wettbewerbs, um die von der US-Regierung immer noch bestehende Behauptung zu widerlegen, dass es Jahre brauchen würde und Rechenleistung im Wert von mehreren Millionen Dollar um eine DES-Nachricht mittels Brute-Force zu entschlüsseln. [25]

4.4 Entwickelte Alternativen

Die EFF schlug schon 1994 dem X9 Kommittee, das für die Empfehlung von neuen Verschlüsselungsalgorithmen verantwortlich war als Alternative triple-DES oder kurz 3DES vor. Damit stellten sie sich klar gegen die NSA und deren Vorschlag eines speziellen Verschlüsselungschips namens Clipper, der einen eingebauten "Backdoor" enthielt, um so der NSA Zugriff auf verschlüsselte Kommunikation zu geben. Ein Backdoor ist eine Hintertür in einem eigentlich als sicher geltendem System. In der Öffentlichkeit nichts von dieser Sicherheitslücke bekannt und sie wird nur von bestimmten Organisationen genutzt. Dabei wurde eine Backdoor im Gegensatz zu einem Exploit durch einen Programmfehler speziell beim Entwerfen des Programms eingebaut.[26] Die EFF schlug dabei 3DES aus den folgenden Gründen vor:

- Grundsätzlich ist DES, auf dem 3DES basiert ein kryptologisch sicherer Algorithmus
- Es eliminiert (für die damalige Zeit) die Möglichkeit einer Brute-Force-Attacke durch den viel längeren Schlüssel, der verwendet wird. Die Schlüssellänge wird zu DES verdoppelt.

24 https://w2.eff.org/Privacy/Crypto/Crypto_misc/DESCracker/ - 12.03.2016 - 15:20
25 https://w2.eff.org/Privacy/Crypto/Crypto_misc/DESCracker/ - 12.03.2016 15:20
26 http://www.itwissen.info/definition/lexikon/Backdoor-backdoor.html - 12.03.2016 - 15:40

- Es ist leicht in bereits vorhandene Systeme für DES zu integrieren

[27]

4.5 Ungeeignete Alternative: 2DES

Da man mit 3DES die Schlüssellänge nur verdoppelt kann man sich fragen, warum man nicht einfach nur 2DES verwendet. Hierbei würde man mit 2 DES Schlüsseln einen Klartext zunächst mit dem 1ten Schlüssel und danach diesen verschlüsselten Text mit dem 2ten Schlüssel verschlüsseln. Da DES keine Gruppe ist würde dies tatsächlich nicht nur einem neuen Schlüssel entsprechen. Allerdings ist dieses Verschlüsselungsverfahren anfällig für die sogenannte Meet-in-the-middle Attacke. [28] Die Meet-in-the-middle Attacke kann nur bei einem Verschlüsselungsalgorithmus mit einem sogenannten Intermediate State funktionieren. Eine kleine Darstellung einer Meet-in-the-Middle Attacke:

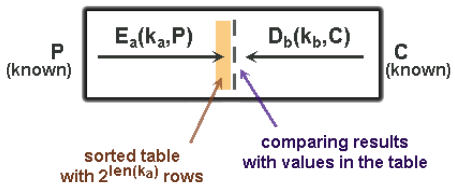


Figure 4: Darstellung einer Meet-In-The-Middle Attacke[29]

Wir brauchen dabei einen Klartext P und den dazugehörigen verschlüsselten Text C. Wir haben nun also:

$$P \rightarrow f(P, k_1) \rightarrow f(f(P, k_1), k_2) \rightarrow C$$

Der Trick ist nun den Zustand $f(P, k_1)$ zu erreichen. Dabei wird zuerst ganz klassisch mit Brute-Force der Klartext P mit allen möglichen Schlüsseln (2^{56} Bit) verschlüsselt und die Resultate gespeichert. Danach macht man das gleiche mit C und der Entschlüsselungsfunktion.

Es gilt also: $P \rightarrow f(P, k_1) = d(C, k_2) \leftarrow C$

Man vergleicht immer mit dem Resultat der Verschlüsselung von P. Es kann dabei mehrere Paare geben die zusammen passen, aber endlich viele. Diese kann man dann ausprobieren ob sie auch bei anderen verschlüsselten Texten funktionieren um so die richtigen 2 Schlüssel herauszufinden. [30]

Größter limitierender Faktor war damals noch der hohe Speicherplatzverbrauch. Man brauchte die Kapazität die gesamten Intermediate Texte zu speichern. Das ist heutzutage kein Hindernis mehr. [31]

Wenn man das Verfahren in 2 oder mehr einfachere Verfahren aufteilen kann, so ist es möglich eine sogeannte "mehrdimensionale MitM Attacke"

27 https://www.eff.org/de/effector/7/14 - 12.03.2016 15:40
28 http://www.nku.edu/~christensen/3DES.pdf - 25.03.2016 - 14:30
29 http://www.crypto-it.net/eng/attacks/meet_in_the_middle.html - 28.03.2016 13:00
30 http://www.crypto-it.net/eng/attacks/meet_in_the_middle.html - 25.03.2016 14:50
31 http://internetofthingsagenda.techtarget.com/definition/meet-in-the-middle-attack - 25.03.2016 15:20

durchzuführen. Dies ist vor allem bei Blockchiffren ein Problem, die auf kleinen Datenblöcken mit sehr großen Schlüsseln agieren. Die Schlüssel werden hierbei effektiv aufgeteilt.

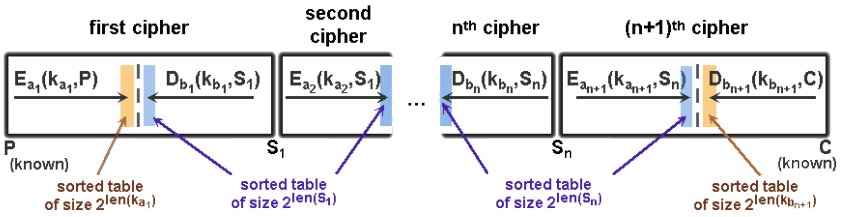


Figure 5: Darstellung einer mehrdimenionalen MitM Attacke[32]

Schlussendlich bekommt man trotz der 2 verwendeten Schlüssel bei 2DES nicht eine resultierende Schlüssellänge von 2^{112} Bit, statdessen braucht man mit Brute-Force nur doppelt so lange um die Nachricht zu entschlüsseln, sollte man im Besitz eines Klartext- verschlüsselter Text Paares sein. Das enspricht 2^{57} Bit. Damit war 2DES nicht zukunftstauglich und wurde nicht in Betracht gezogen als Alternative für DES. [33]

4.6 triple-DES

3DES wurde im Standart X9.52 festgehalten und wurde zur neuen empfohlenen Verschlüsselungsmethode in der FIPS, der Federal Information Processing Standard. Dieser regelt die Standards, die Firmen erfüllen müssen um Verträge mit der US-Regierung zu schließen. Dies trug maßgeblich zu Verbreitung des 3DES in viele mit der Regierung zusammenhängenden Wirtschaftszweige bei. Zugleich wurde Single-DES oder DES nur noch für wenige, speziell zugelassene "legacy systems" erlaubt. Das sind Systeme die zu alt und groß waren, und die eher intern genutzt wurden.
Zudem wurde die langsame Umstellung auf AES empfohlen, da dieser auch von der NSA speziell authorisiert war. Zunächst wollte man aber die hardwarebasierten DES-Systeme weiter nutzen, deswegen war 3DES ein logischer Schritt.[34] 3DES basiert auf dem DES Algorithmus. Um etwas mit 3DES zu verschlüsseln nimmt man 2 DES Schlüssel mit jeweils 56 Bit und wendet sie wie folgt an:

- Verschlüssele den Klartext mit dem ersten Schlüssel
- Entschlüssle die Nachricht mit dem zweiten Schlüssel
- Verschlüssle die Nachricht nochmal mit dem ersten Schlüssel

Diese besondere Anrt der Anwendug der 2 Schlüssel bedeutet eine Verlängerung des insgesamten Schlüssel auf 112 Bit. Dadurch ist er sehr viel

32 http://www.crypto-it.net/eng/attacks/meet_in_the_middle.html - 28.03.2016 13:10
33 http://www.nku.edu/~christensen/3DES.pdf - 25.03.2016 15:00
34 http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf - 12.03.2016 16:50

schwerer mit Brute-Force zu knacken.[35] Es hat keinerlei Auswirkungen auf die Sicherheit des Verfahrens ob man in Schritt 2 und 3 ver-oder entschlüsselt. Aber in der oben genannten Reihenfolge ist es üblich. Man kann nur nicht immer den gleichen Schlüssel benutzen.[36]

Ein Nachteil von 3DES im Gegensatz zu anderen Verschlüsselungsverfahren ist seine Performanz. Da auf Hardwareverschlüsselung optimiert, ist die dreimalige Anwendung des DES-Algorithmus in Software form deutlich langsamer als von vergleichbaren Algorithmen mit ähnlich hoher Sicherheit. Deswegen ist 3DES im Privatgebrauch auch kaum verbreitet. Hier hat sich vor allem AES durchgesetzt wegen seiner sehr reduzierten Codebasis.[37]

35 http://page.math.tu-berlin.de/~kant/teaching/hess/krypto-ws2006/des.htm - 12.03.2016 16:20

36 http://www.itwissen.info/definition/lexikon/triple-DES-3DES-Dreifach-DES.html - 12.03.2016 16:30

37 http://www.netplanet.org/kryptografie/verfahren.shtml - 12.03.2016 16:30