Chiffrendesign

Will man ein Verschlüsselungsverfahren entwickeln, bieten sich zwei Wege an:

- Man macht das Verfahren möglichst kompliziert und hofft, dass dadurch keine Schwachstellen entstehen – oder dass ein Angreifer diese in der Komplexität nicht findet. Diesen Ansatz nennt man auch "Security by Intricacy" (in etwa "Sicherheit durch Unduchschaubarkeit")
- Man entwirft das Verfahren möglichst durchdacht und versucht, Schwachstellen gar nicht entstehen zu lassen.

Wenig verwunderlich kommt bei ernsthaften Verfahren nur die zweite Variante zum Einsatz.

Was ist eine Schwachstelle eines modernen Verfahrens?

Zunächst muss man sich klarmachen, was man bei modernen Verfahren unter einer Schwachstelle versteht.

Bei einem modernen Verfahren spricht man bereits dann von einer "Schwachstelle", wenn es einen Angriff auf das Verfahren gibt, der besser ist als die vollständige Schlüsselsuche.

Außerdem werden statistische Auffälligkeiten bei einem modernen Verfahren als Schwachstelle betrachtet, da diese als Angriffpunkt dienen können. Ein gutes symmetrischen Verfahren soll ein **Zufallsorakel** sein.

Als Zufallsorakel bezeichnet man eine Funktion, bei der kein erkennbarer Zusammenhang zwischen der Eingabe (Klartext & Schlüssel) und der Ausgabe (Geheimtext) existiert – auch nicht in wenigen Einzelfällen.

- Wenn ein Verschlüsselungsverfahren beispielsweise bei Verwendung des Schlüssels `00…000` stets einen Geheimtext liefert, der als letztes Bit eine `0` hat, ist die Zufallsorakel-Eigenschaft schon verletzt.
- Ebenso hat man kein Zufallsorakel mehr, wenn das Invertieren von Klartext und Schlüssel dazu führt, dass auch der Geheimtext invertiert. Dies ist beim DES der Fall. Unter anderem deshalb ist der DES kein perfektes Zufallsorakel.
- Wenn es Schlüssel gibt, bei denen Verschlüsselung und Entschlüsselung identisch sind - dann führt das zweifache Verschlüsseln wieder zum Klartext. Das ist bei DES bei einigen wenigen Schlüsseln der Fall → kein Zufallsorakel.

Überlegungen zur Schlüssellänge

Schlüssellänge	Anzahl der Schlüssel	Dauer einer vollständigen Schlüsselsuche	
40 Bit	1,1·10	**12**	1,3 Sekunden

56 Bit

7,1·1016

24 Stunden

64 Bit

1,8·1019

256 Tage

80 Bit

1,2·1024

45.965 Jahre

128 Bit

3,4·1038

1,3·1019 Jahre

192 Bit

6,3·1057

2,4·1038 Jahre

256 Bit

1,2·1077

4,4·1057 Jahre